BACHELORARBEIT

im Studiengang Audiovisuelle Medien

Conception and development of a mobile mixed reality medium for environment related storytelling

– A novel approach to virtual heritage –

Vorgelegt von Jessica Bergs
an der Hochschule der Medien Stuttgart
am 24.03.2014

Erstprüfer: Prof. Dr. Simon Wiest
Zweitprüfer: Prof. Jörn Precht

Hiermit versichere ich, Jessica Bergs, an Eides Statt, dass ich die vorliegende Bachelorarbeit mit dem Titel: “Conception and development of a mobile mixed reality medium for environment related storytelling – A novel approach to virtual heritage” selbstständig und ohne fremde Hilfe verfasst und keine anderen als die angegebenen Hilfsmittel benutzt habe. Die Stellen der Arbeit, die dem Wortlaut oder dem Sinn nach anderen Werken entnommen wurden, sind in jedem Fall unter Angabe der Quelle kenntlich gemacht. Die Arbeit ist noch nicht veröffentlicht oder in anderer Form als Prüfungsleistung vorgelegt worden.

Ich habe die Bedeutung der eidesstattlichen Versicherung und die prüfungsrechtlichen Folgen (§26 Abs. 2 Bachelor-SPO (6 Semester), § 23 Abs. 2 Bachelor-SPO (7 Semester) bzw. § 19 Abs. 2 Master-SPO der HdM) sowie die strafrechtlichen Folgen (gem. § 156 StGB) einer unrichtigen oder unvollständigen eidesstattlichen Versicherung zur Kenntnis genommen.

Ort, Datum Unterschrift

ABSTRACT

The goal of this thesis is to develop a novel type of virtual heritage medium that utilises the combined immersive and engaging potentials of interactive mixed reality environments and spatial narratives.

Concretely, this is achieved through depth-sensitive compositing of real-time 3D content into the live-video of a tracked smartphone. The user can explore this mixed reality environment, watch the actions of staged 3D characters as well as interact with them and virtual artifacts. This medium would therefore provide possibilities for telling stories in direct context with existing environments along with an immersive and engaging media experience. This work will mainly focus on how this medium can be used as an edutainment medium in sites of cultural heritage.

This thesis will focus on establishing the technical requirements and realisation possibilities for implementation in Unity on iPhone 5 / iOS 7. Subsequently, a prototype is implemented in order to prove the research results.

Keywords: Mixed reality, mobile computing, spatial storytelling

KURZFASSUNG

Das Ziel dieser Arbeit ist die Entwicklung eines neuartigen Virtual Heritage Mediums das User mit Hilfe von interaktiven Mixed-Reality Umgebungen und raumbezogenem Erzählen nicht nur räumlich mitten in eine Geschichte hineinversetzt, sondern auch aktiv in diese einbezieht.

Dies wird erreicht, indem das Videobild eines getrackten Smartphones mit perspektivisch stimmigen Echtzeit-3D Inhalten überlagert wird. Der User kann diese Mixed-Reality Umgebung erkunden, die Handlungen von 3D Charakteren beobachten sowie mit ihnen und virtuellen Artefakten interagieren. Dieses Medium bietet folglich die Möglichkeit mediale Geschichten in echten Räumen zu erzählen sowie ein immersives und user-involvierendes Medienerlebnis. Diese Arbeit wird den Einsatz dieses Mediums speziell für Kulturvermittlungszwecke fokussieren.

Diese Arbeit wird zunächst die technischen Anforderungen und Umsetzungsmöglichkeiten dieses Vorhabens mittels Unity für iPhone / iOS 7 untersuchen. Die Belegung der Ergebnisse erfolgt durch die anschließende Realisierung eines Prototypen.

Schlagwörter: Mixed reality, mobile Medien, Storytelling im Raum

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

CV	–	Computer vision
CGI	–	Computer-generated imagery
COTS	–	Commodity off-the-shelf
DGPS	–	Differential global positioning system
GUI	–	Graphical user interface
HCI	–	Human-computer interaction
IrMoCap	–	Infrared Motion Capture
LOD	–	Level of detail
MEMS	–	Micromechanical systems
MP	–	Megapixel
MR	–	Mixed Reality
MRE	–	Mixed Reality Environment
MRS	–	Mixed Reality System
RTLS	–	Real time localisation system
SfM	–	Structure from motion
TOA	–	Time of arrival

1 INTRODUCTION

1.1 PROBLEM STATEMENT

"Virtual heritage" has established itself as a widely used edutainment medium for fostering public understanding and appreciation of cultural heritage (The ICOMOS Ename Charter for the Interpretation of Cultural Heritage Sites 2008; Mosaker 2001).

These virtual environments provide architectural reconstructions of a historic site, e.g. of a specific time period. Visitors can navigate through this 3D model from a first-person perspective and are thereby given the opportunity to immersively explore the past place themselves.

Many critics, however, argue that although virtual heritage offers great visualisation possibilities, it misses its potential of conveying cultural significance and sufficiently engaging the visitor into a learning experience (Tan and Rahaman 2009):

Mosaker (2001) states that only an architectural reconstruction itself does not provide visitors cultural information of the past as it lacks those essential elements that actually define its cultural background: The everyday life of people, the important events in that place in that time. "After all, the buildings were made by and for their inhabitants" (Mosaker 2001:4). Champion (2002) therefore argues that virtual heritage should rather be designed as a "virtual environment that people with a different cultural perspective occupy […] as a 'place'" (Champion 2002:3). Concretely, such a virtual population could be used to simulate their day-to-day life in past times or to reproduce site-related historical events, providing visitors with highly relatable cultural information.

Furthermore, virtual environments are also criticised for being little engaging: Even if visitors are provided with the interaction possibility of exploring the site themselves, they will feel "lost and bored" after some time if they are not given other interaction possibilities or a specific goal to pursue (Tan and Rahaman 2009:6). It has therefore been suggested to enrich the virtual environment with meaningful interaction possibilities, e.g. as used in serious games, as incentives for visitors to further engage with the virtual environment and to amplify the learning effect (Champion 2002:6).

Thanks to the advances of modern technology, adequate tools have become available for realising such a novel type of virtual heritage environment. With the rise of augmented reality and mobile computing, also the possibility has opened to use a site's environment itself as the very stage for the virtual population and placing the user right in the middle of the happenings. As a result, the visitor would not only be provided a culturally rich and engaging but also a highly immersive edutainment medium.

1.2 CONCEPT

This thesis proposes to utilise the combined immersive and engaging potentials of interactive mixed reality environments and spatial narratives for a novel type of virtual heritage experience.

Concretely, this is to be achieved by developing a mobile mixed reality medium that facilitates interactive and real-environment related storytelling.

By the use of the medium, visitors of a heritage site are provided an illusory experience of finding themselves surrounded by virtual characters and artefacts populating the real-world space around them. This mixed reality environment (MRE) can serve for (re-) staging events and situations related to the site's historical background. The visitors can explore this newly vivified space and watch the scenario's plot unfolding. Equipping the virtual elements with interactive behaviours opens the possibility to incorporate the user into the storytelling and thereby actively engage him in its shaping.

This mixed reality system (MRS) would therefore provide new possibilities for telling interactive stories related to existing environments along with a highly immersive and contextualised virtual heritage experience.

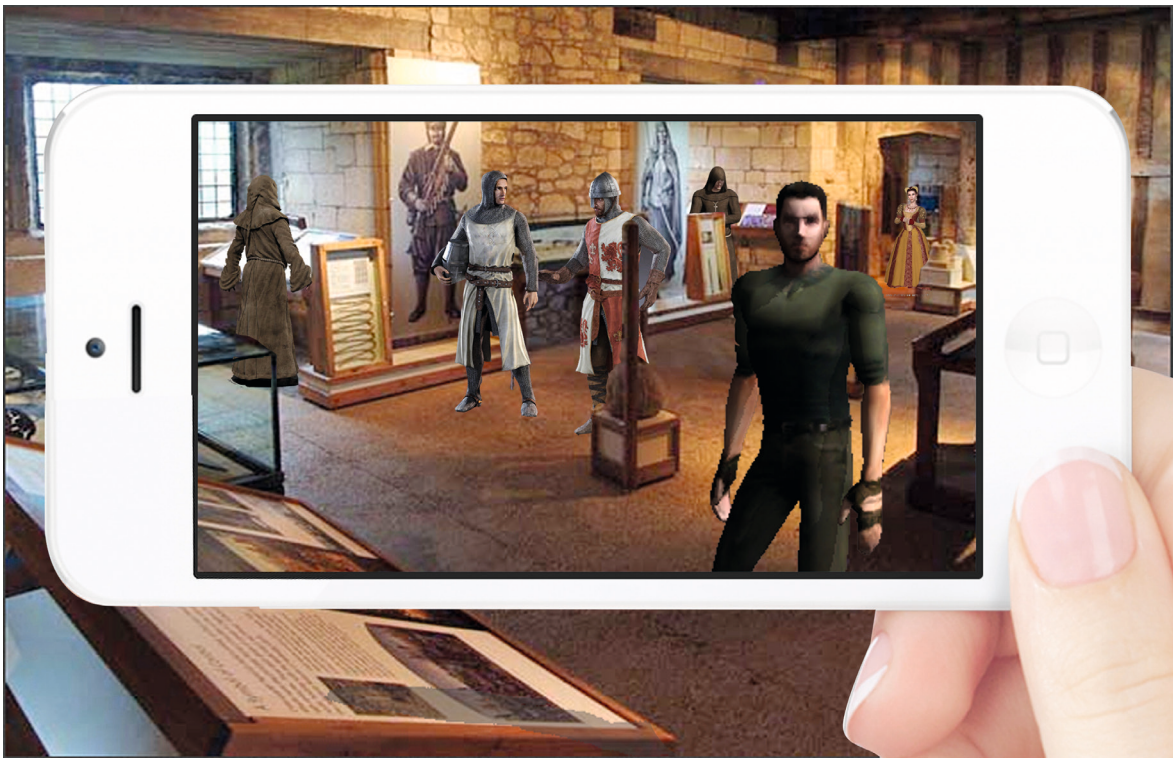

Fig. 1: Concept key visual

1.3 TARGETED TECHNICAL PRINCIPLE & BASIC SYSTEM ARCHITECTURE

1.3.1 Targeted principle

The visitor will be equipped with a smartphone as the main hardware device that serves him as a “viewing window” when exploring the MRE. The smartphone provides mobile computing power, a display, headphone interface and backside camera.

1.3.1.1 Training phase

The smartphone has been prepared with an app providing a real time 3D environment, in which the site has been rebuilt as a 3D version. In this virtual replica, further 3D objects and sounds have been “staged” and added interactive behaviour according to the scenario’s storytelling concept. Lastly, a virtual camera was added to the scene.

1.3.1.2 On runtime

This virtual world camera is wired to a tracking system that monitors the location and rotation of the real-world’s smartphone camera while the visitor is moving through the site. As a result of this matching process, both cameras show the same view of the scenery in the real and in the virtual site version. For achieving the depth-sensitive compositing of the two cameras’ video streams, the virtual site replica is used as an occlusion model, cutting out those parts of the storytelling related objects that would be hidden behind real-world ones. The prepared CGI stream is now superimposed upon the smartphone camera’s video stream and the resulting composited video is output to the visitor’s display.

1.3.2 System architecture

Above targeted principle will be implemented for use on an iPhone 5 with iOS 7, representing an exemplary smartphone specimen. This mobile device offers all required technical components at state-of-the-art quality. Moreover, using such a widely distributed technology should support this project’s purpose of showing that advanced MR experiences do not necessarily require lab technology but can already be delivered on commodity-of-the-shelf devices.

The real time 3D environment will be realised via the real time game engine Unity, as it offers the all the required functions and tools, e.g. asset management, AI, game physics, 3D object creation, texturing & shading (Iuppa 2006:188).

1.4 RESEARCH OBJECTIVES & THESIS OUTLINE

So far, the MRS’s general concept, targeted usage area and technical principle have been outlined. The purpose of this thesis is to examine how this system can concretely be realised in a way that it delivers a compelling MR experience while requiring – preferably – nothing more than an iPhone 5.

As a first step to achieving this goal, it will have to analysed what this system minimally requires in order to deliver an effective result and to what extent its possibilities are limited by the smartphone’s technical capacities. It will have to be analysed which implementation possibilities are available that meet these frame conditions as well as how the they can be used most efficiently for delivering maximum results with the given resources. The resulting solution should provide an MR authoring framework that can be used for designing environment-related storytelling scenarios.

In order to answer above questions, this thesis addresses the following research objectives in the listed order:

- Developing an effective & efficient technical realisation approach for an MR authoring framework system, based on Unity for use on iPhone 5 by
    - analysing those technical minimum “must” demands of the MRS that are inevitably essential for achieving the targeted MR experience
    - quantifying the given limitations of the basic set-up and evaluating their implications on the system’s design and the maximum achievable result
    - based on these framings, analysing and evaluating solution possibilities that meet the stated requirements and achieve the best possible result within the given limitations
- Proving the research results by implementing a prototype

1.5 EXPECTED RESULTS & SUCCESS CLASSIFICATION

The following cases provide an outlook on the possible outcomes of the finalised approach as well as an assessment scheme for the project’s success:

- Best case: The MRS works and is implemented all “in-phone”: The system could be fully implemented within the mobile device and delivers acceptable results. No further hardware or other aiding set-ups in the room are necessary.
- Middle case: The MRS works, but external hardware required: The MRS requires additional external hardware and/or aiding set-ups in the room in order to deliver acceptable results.
- Worst case: The MRS does not work. The system cannot be implemented or neither in-phone nor external components can achieve acceptable result

2 ANALYSIS OF BASIC REQUIREMENTS & CONSTRAINTS

The challenge of this project is to find a realisation solution that fully meets both the concept's demands: Delivering the MRE's illusion – using the technical resources of a smartphone. Moreover, the medium should be designed in a way that it is easily "usable" for the targeted site visitors, i.e. using it should not require technical expertise.

These demands consequently state the frame conditions for the MRS' design possibilities. This section will quantify them by applying the following top-down approach:

First, it will be assessed which immersive and interactive features the MRE will need as an absolute "must" for triggering the targeted illusion in the user. This minimum set of features will then be a fix part of the HCI-design base accompanied by usability considerations. It will further have to be examined what technical demands they impose on the system's architecture. These requirements form the lower limit the final MRS will definitely have to meet in order to deliver an effective result.

The upper limit to what is achievable is set by the constraints of the technical resources' capacities and characteristics. These will have to be quantified as well in order to precisely assess the maximum result the MRS "can" achieve.

As a summary, the analysis' results will be broken down into the technical frame conditions for the individual system components. In the subsequent development process, they form the basic evaluation criteria for the analysis of available implementation possibilities.

2.1 MINIMUM REQUIREMENTS

2.1.1 Conditions of the illusion of presence & engagement

The final MRE should provide the user an illusion of being surrounded by virtual happenings and to be actively engaged in them.

Sanchez-Vivez and Slater (2004:4) state that this "feeling of being and acting in a virtual world" is the product of our cognitive processes with which we respond to the immersive and interactive attributes of mediated environments.

This cognitive phenomenon can be so powerful that it prompts users to react to a mediated environment as if it was real – although they perfectly know it is not (Slater 2009; Sanchez-Vives and Slater 2004:18). "No one could ever be fooled into believing in the reality of any virtual environment that is capable of being displayed in real-time with today's equipment" (Sanchez-Vives and Slater 2004:15). It is therefore important to mention that the targeted illusion does not aim at being mistaken as "reality" but at being believable enough that users embrace it.

According to Slater (2009) above response is controlled by two parameters: Place illusion (PI) and Plausibility Illusion (PsI).

PI stands “for the type of presence that refers to the sense of ‘being there’”. It is given “if a person perceives the virtual world making use of motor actions to perceive in the same way as perceiving the real world” and as long as no stronger intrusive “cues” indicate otherwise (Slater 2009:4 sqq.).

Achieving “place illusion” consequently requires the following features: The depiction of virtual space has to resemble certain characteristics of the real world in order to be perceived as a “place”. Two examples for such characteristics are the representation of perspective and light-shadow correlations. Regarding mediated reality environments, this implies that the spatial depiction of virtual objects has to be consistent with their real-world surroundings. Furthermore, the more freely and naturally the user can explore this “place” with his own motoric actions (e.g. walking), the higher the quality of the PI. (Sanchez-Vives and Slater 2004:10). This likewise implies that when the user moves within a room, the simulation’s visual update should match to his proprioception, e.g. if he turns his head also the virtual camera should turn in to the desired view. Moreover, for being perceived “as if real”, the virtual world should be shown at a human-like field of view. While exploring the MRE, the user’s field of vision should not be intruded by unmediated reality as this would state a constantly reminding “cue” that he is actually not in the simulated place but still in the here and now. (Sanchez-Vives and Slater 2004:4).

Consequently, the PI conditions are the driving forces for evoking the feeling of presence and are addressed by the immersive features of mediated environments.

PsI refers to the extent that a mediated environment behaves “logically” to the actions of the user. This implies that the user’s actions have to result in those consequences that he would expect in real life (Slater 2009:9; de la Peña et al. 2010:4). This action-reaction correlation is implemented by means of interaction, which gran the user the ability to deliberately influence the happenings around him.

PsI consequently forms the basis of engagement and is facilitated by the interactive features of mediated environments.

Overall concluding, the final quality of the illusion cannot be ensured or quantified, for being dependent on the individual user and therefore a subjective matter.

It can be assumed, though, that if the conditions of PI and Psi are met by – objectively measurable – highly immersive and interactive elements of the MRE (Sanchez-Vives and Slater 2004:4), the user’s cognition should respond with the targeted feelings .

2.1.2 Resulting HCI design concept

Based on above findings, the following section will outline the resulting human-computer interaction design concept stating all those features that the MRE will have to provide in order to facilitate the targeted user experience and usability (Preece, Rogers, and Sharp 2002:12, 44).

User experience design (UxD) aspects aim at making an interactive product “pleasurable”. Related aspects include play, interactivity, engagement, and style of narrative. (Preece, Rogers, and Sharp 2002:50). The MRE’s targeted user experience should manifest in the evocation of the feeling of presence and engagement in the user, the conditions to which have been analysed in the previous section. The MRE’s user experience design considerations will hence be concerned with examining how the conditions of PI and PsI can concretely be translated into corresponding immersive and interactive features. Moreover, it will be assessed what technical requirements the implementation of these features state for the overall MR system.

The more objective usability aspects (Preece, Rogers, and Sharp 2002:12, 44) concern those features that are necessary for making the final medium effectively usable for the target group, in this case visitors of cultural heritage sites. Also here a concrete design concept will be provided at the end of the section.

2.1.2.1 User experience design aspects

I) Components of achieving PI

Achieving PI (place illusion) requires that the user can perceive the MRE in a way as he does his real environment. This is to be provided for by the following set of immersive features:

I.I) Consistent depiction of the virtual in the real world

In order that the MRE is perceived as a coherent entity, the virtual objects have to show a certain consistency with their real world surroundings in wich they are placed. The quality of this feature is controlled by several parameters:

- ***Extent of knowledge of the real world environment***

  The targeted MRE mixes the realities in a way of spatially placing virtual object within a real world environment (a virtual object "lies" on the floor of the room and is partly occluded by a real-world pillar) and equally puts both worlds into meaningful context (a virtual book "lies" on a real-world table, virtual characters sit on real-world benches).

  Achieving this requires that the virtual world has a certain "extend of knowledge" of the real world, which includes the aspects 'where' objects are (the occluding pillar) as well as 'what' they represent, i.e. their meaning or use (a bench to sit on). (Milgram and Kishino 1994:9).

  It has been stated in the basic concept that the MRE will be "furnished" with a virtual replica of the real room. But it still has to be cleared, though, to what precise extent this modelhas to reconstruct reality: Which objects will definitely have to be included and at what level of detail and which of these will definitely have to be assigned "meaning" for the virtual population?

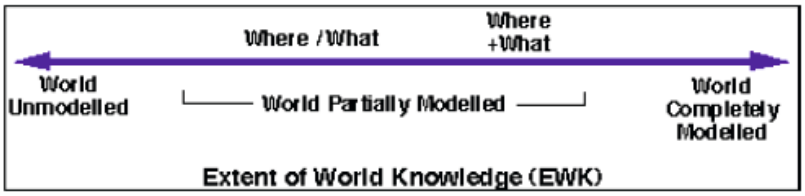

Fig. 2: EWK continuum (Milgram and Kishino 1994:9)

For its purpose as an occlusion model, the model should definitely include all those objects in the room that could occlude the users view on a virtual object placed behind it. Also all objects that are generally relevant for the virtual population have to be included, e.g. stairs and walls. Regarding the taxonomy of Milgram & Kishino, this covers the 'where' aspect. All these objects need to be digitalised precisely enough to later match their real-world equivalent reliably. Otherwise, if an imprecise models' edges cut off too much or too little an occluded virtual objects, imprecise registration occurs. This would clearly compromise the targeted illusion. Also the objects' relative location to each other has to be reproduced precisely for the same reason.

As regards the 'what' aspect, the virtual population first and foremost needs to "know" that these real world objects are rigid obstacles: The characters will have to walk around objects standing in their way and have to halt in front of walls. For objects of "use", e.g. the benches

mentioned, the characters could be equipped with animations that if they stand around such an object, they will sit down on it. This aspect also aids to the plausibility illusion, as the population should not only react "logically" to the user's actions but also to the real-world environment in general.

- ***Accurate registration***

  A crucial factor to the credibility of the MRE is the fitting accuracy of the registration, i.e. the precise superimposition of the virtual upon the real-world video (Lima et al. 2010:1).

  As Azuma (1997) argues, the human eye already detects tiny offsets and geometric inconsistencies. Therefore even small imprecisions would compromise the illusion of ‚that virtual object is lying on top of that real table.' Consequently, it is vital for the PI that the registration accuracy is as high as possible.

  Offsets can have multiple causes, all of which the final MRS should seek to avoid: For one, offsets can result if the optical characteristics of the real camera and the virtual camera do not match. This includes optical distortion parameters, viewport dimensions, field of view, translation and orientation parameters, etc.

  Another major error source can be caused by shortcomings of the tracking system, e.g. if it does not support a sufficient accuracy standard, responds with a high error rate or even tracking breaks. Also tiny time gaps between the tracking and the visual update lead to offsets (Azuma 1997:18 sqq.).

  Of course, also inaccuracies in the virtual reproduction of the real room inevitably lead to offsets, may it be their positioning or detail level.

- ***Deliberate degree of reproduction fidelity***

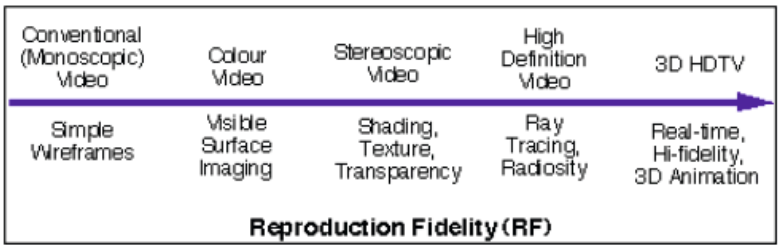

Fig. 3: RF continuum (Milgram and Kishino 1994:11)

The "reproduction fidelity" dimension of the MR-taxonomy (Milgram and Kishino 1994:11) puts two factors to the quality of the final visual output:

The first one concerns the outputting display, including characteristics like resolution, brightness / contrast ratio and colour gamut. For the MRE, the display should first and foremost provide a sufficient resolution in order to present the output at a decent LOD. Starting from here, the rule of "the more the better" can generally be applied.

The second factor regards the graphics quality:

For achieving a homogeneous look of the MRE, the appearance of the virtual objects should show the following graphical consistencies with the "real world" view: First of all, both video streams should share the same contrast, brightness, dynamic range and colour gamut as the very basis for blending them (Naimark 1991:2). In addition, the virtual objects should show spatial consistency, i.e. that they are depicted with the same focus range as other real objects of that distance (Azuma 1997:16). They should further be subject to lighting and shadow conditions of their environment.

The virtual props should be basically recognizable as the objects they represent. Their mesh and texture should therefore be of a sufficient LOD.

As far as more visual realism of the virtual objects is concerned, Slater, Mosaker and Tost and Champion (Slater 2009:6; Sanchez-Vives and Slater 2004:7; Mosaker 2001:8; Tost and Champion 2007:3) all agree that the degree of presence is not much further enhanced if the virtual objects looked even more "real." Iuppa (2006:161) and Mosaker (2001:8) even argue that a too high level of visual realism might even distract the user from the essential parts of the simulation.

Concluding, the blending quality should minimally target an overall homogeneous look. If the final MRS should provide more rendering capacities, these could be used for more visual realism of the virtual objects, though this would rather aid to the overall aesthetics of the MRE than to the place illusion.

I.II) Field of view

For mimicking the human's visual perception, the MRE should provide a corresponding field of view: The human full field of vision spans ca. 180 horizontal and ca. 120° vertical while the central field of vision is reduced to ca. 50-60° horizontal and ca. 55° vertical (Stockyard Hill Wind Farm 2009:3). The cameras should therefore target to provide the full field of view in order to mimic human perception best. If this is not possible, they should at least provide the central field of vision as a secondary option.

I.III) Display size

Regarding the displays size, if the user can still see parts of unmediated reality while exploring the MRE, these parts intrude his illusion of being "surrounded" by the virtual content. (Sanchez-Vives and Slater 2004:2; Slater 2009:3). Consequently, the display should cover as much of the user's field of vision as possible.

I.IV) Portability

In order to allow the user to walk freely through the room while exploring the MRE, he should not be restricted in any way, e.g. by cables or a limited "activity area".

Slater (2004:10) and Toast et al. (2007:9) have shown in their case studies that their probands stated to have felt more present if they were less restricted by the hardware and if they could use natural input possibilities. Concluding, the system should not feature any cable-bound components and it should be capable of reliably tracking the user over the whole room.

Additionally, the system should be robust enough to cope with a comfortable handling speed, i.e. that the users should not be bound to move the handheld device unpleasantly slow.

I.V) Temporal continuous proprioception

The user should be enabled to explore the MRE in a way that matches his proprioception: While he moves the MRS's handheld component, its visual output should reproduce his movements as continuously and synchronously as if real.

Achieving the illusion of a continuous and smooth movement on screen requires a minimum update rate of 15 fps (Naimark 1991:8), while frame rates of 24 fps are standard practice in the film industry. The MRS's graphics update should therefore operate at an average frame rate of 24 fps and never below 15 fps as the absolute minimum.

The proprioceptive movement reproduction equally relies on the tracking data update rate, which consequently has at least to match that of the graphics.

Ensuring that the updates remain synchronous to the user's movements throughout the experience makes stable real-time performance of the overall system a critical factor (Naimark 1991:8). More precisely, both the graphics and tracking updates have to perform in real time (Sanchez-Vives and Slater 2004:7) Should one of these factors fail to do so, the resulting discontinuities would make the MRE seem to lack behind or stall, breaking the proprioceptive response (Sanchez-Vives and Slater 2004:2 sq.). Consequently, for achieving a consistent frame rate of between 15 and 24 fps throughout in real time, the overall system latency has to range between average 42 ms and maximum 66 ms per frame.

II) Components for achieving Psi

II.I) Providing basic physical interaction-reactions

Making the MRE behave "realistically" to the user's actions requires first and foremost that the user is actively incorporated into it as a "physical" object. This implies that he should not be able to simply walk "through" virtual elements, but that they react to his contact in a way he would expect in real life.

For achieving this, virtual objects will have to be equipped with interactive behaviour that is related to what they represent.

Not only should loose objects in the room move aside if the user walks against them, but a small cube with a granite texture should also move differently than one with a cardboard texture. Virtual characters should step aside if the user enters their artificial "comfort zone". Linking these visual reactions with corresponding sound effects would render them even more credible.

When considering to add virtual props that are actually not "moveable" (e.g. extra walls, stairs, large granite blocks) it should be kept in mind that the MRE's "physical cause-effect correlation" (Slater 2009:9) is a single-sided matter: While the user can be equipped with a physical body valid in the virtual world, virtual objects cannot exert "real" physical force on the user. A virtual wall cannot function as an obstacle for the user as he can simply walk through it. Such object-types should therefore preferably be avoided or be used rather as decorative than functional elements.

This is the most basic measure for incorporating the user into the MRE. There are numerous other interaction possibilities for facilitating far more engagement, e.g. communicating with social agents or letting the user's action trigger off certain events. Their use, however, is not universal but rather depends on the individual scenario's story and design. Therefore, these possibilities will be discussed in the storytelling-related part of this thesis (see "3.1.2.3 Staging and interaction possibilities for storytelling").

2.1.2.2 Usability aspects

As the term implies, usability aspects are concerned with ensuring that an interactive product is not only ‚useable' but also easy to use, i.e. that it should help the user to carry out his tasks effectively and efficiently. This also includes that it should be easy to learn and memorise how to operate the system (Preece, Rogers, and Sharp 2002:45, 50; Bevan 2001:2).

Good usability is of fundamental importance for the MRS: No matter how compelling the quality of the MRE, no matter how lifelike the interaction with its population, the user will resent the app in frustration if he is constantly left to watch it slacking or freezing – particularly if this happens without any further (comprehensible) information or recovery options.

For putting above goals to practice, Nielsen (2001) defines a set of usability principles, also referred to as heuristics:

- “Aesthetic and minimalist design-avoid using information that is irrelevant or rarely needed
- Error prevention-where possible prevent errors occurring in the first place
- Visibility of system status-always keep users informed about what is going on, through providing appropriate feedback within reasonable time
- Match between system and the real world-speak the users’ language, using words, phrases and concepts familiar to the user, rather than system-oriented terms
- Help users recognise, diagnose, and recover from errors-use plain language to describe the nature of the problem and suggest a way of solving it
- User control and freedom-provide ways of allowing users to easily escape from places they unexpectedly find themselves, by using clearly marked ‘emergency exits’
- Recognition rather than recall-make objects, actions, and options visible
- Help and documentation-provide information that can be easily searched and provides help in a set of concrete steps that can easily be followed and provides help in a set of concrete steps that can easily be followed”

As regards the MRS, these principles can be approached as follows:

- Designing the system’s error handling as “self-sufficient” as possible:
    - It goes without question that the system should generally work robustly and steadily in the first place.
    - In order to not disturb the user’ s place illusion, the system status should stay “invisible”, i.e. not display constant (text) info about the system status. Regarding minor error handling, this also implies that the system should generally be able to recover fast and independently without the user noticing.
    - Yet, should problems arise that lead to such noticeable performance sags or even failures that would break the place illusion anyway, the user should be informed about the problem in short and simple terms. If his intervention is necessary, he should be instructed how to support the recovery by little and easy means. For example, if the tracking system breaks, the user should be informed by a simple graphic how to reinitialise it, e.g. by simply slowing down his movement.
- Providing a main menu with resume, help and restart option:
    - The MRE-scenario should also provide a permanent option for navigating back to the main menu. This should feature a “resume” option for getting back to the scenario where the user left it, a “help” text for basic use instructions as well as a restart option, so that the user can restart the scenario manually. This might be necessary e.g. in cases of noticeable run-time errors that were not detected by the system.

2.2 RESOURCE LIMITATIONS TO BE CONSIDERED

2.2.1 iPhone 5 mobile computing capabilities

The iPhone 5 uses an Apple A6 ‚system on a chip' with a 32-bit ARMv7 dual core CPU delivering 1.3 GHz computing power and a 266 MHz Power VR graphics processor (Phone Arena 2013). The phone has 1 GB RAM DDR2 memory, which is shared between the CPU and the GPU (Wiebe 2011:58).

Comparing these capabilities to those of a common workstation of the computer graphics industry, e.g. an Apple iMac with 64-bit 2.9 GHz quad-core CPU, 16 GB DDR3 memory plus 1 GB dedicated graphics memory (Apple 2013a), the rather limited processing power of the smartphone becomes clear.

The shared CPU/GPU memory implies that if the CPU load is high, e.g. caused by the tracking system, this might affect the graphic rendering update rate and vice versa (Wiebe 2011:58).

Considering these specifications, the following factors for the MRS can be concluded:

- The MRS will generally have to be designed in a such a "cost-efficient" way that its total processing and memory costs are permanently within the range of the phone's maximum capacities. Only then the required stable real-time performance can be ensured.
- Even within this 100% range: Due to the shared memory for tracking and rendering, it has to be actively avoided that both costs affect each other in a way that either the latency or the graphics rate falls below required. As stated in the section above, the MRS requires both rates to be of permanent real-time frequency.
- As the MRS can neither compromise on the tracking system's registration accuracy nor on its update rate, this component has to be given the priority over the graphics. Concretely, this means that the tracking system is given all the memory space it requires for achieving acceptable results. The quality of the graphics/story-interaction components will then have to be cut back until they "fit into" the remaining space while still working at real time. This can e.g. be done by reducing the overall 3D objects' vertices count, use lower resolution textures, cheaper shaders, use less complex animations, etc.

2.2.2 iPhone 5 camera specifications & image quality

2.2.2.1 Sensor

The iPhone 5s' iSight camera (back side) provides a 1/3.2" (4.54 x 3.42 mm) sensor of 3264 x 2248 pixel (7.33 MP, aspect ratio 4:3), operating at 50 ISO standard sensitivity.

According to the camera's EXIF data, the lenses' focal length is f 4.12 mm at a crop factor of 7.61, which equals f 33 mm on a 35 mm camera or a light wide angle lens, respectively.

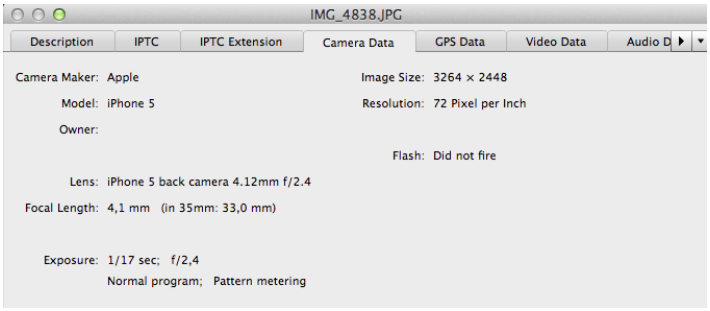

Fig. 4: iPhone 5 camera photo EXIF data

The camera's horizontal and vertical field of view (FOV) α can be calculated from its focal length f and its respective sensor dimension *d* (Basset 2013):

$$\alpha = 2 \arctan \frac{d}{2f}$$

The camera consequently offers a field of view of 57.42° horizontal and 45.50° vertical, slightly less wide than the human central field of vision of approximately 50 - 60° horizontal and 55° vertical (Apple 2013b).

2.2.2.2 Video options & FOVs

The camera records Full HD video (aspect ratio 16:9) at 30 fps (Apple 2013b). The raw video stream uses the full width of the screen, resulting in 3264 x 1836 pixel resolution and a diagonal FOV of 57.42° h / 34.23° v.

The camera also supports video image stabilisation, by the help of which minor camera shakes can be balanced. This is achieved by allowing a slightly smaller image section of 2792 x 1569 pixel to float over the sensor. The difference between an unstabilised and stabilised video is definitely noticeable. This makes it a useful feature but results in an even narrower FOV of 50.26° h / 29.38 v. The image stabilisation is active by default but can be accessed via the iOS AV Foundation framework.

Full sensor	Raw video stream, unstabilised	Raw video, stabilised
3264x2248 pixel	3264x1836	2792x1569
4:3	16:9	16:9
FOV diagonal 69.10°	FOV diagonal 65.37°	FOV diagonal 56.45°
FOV horizontal 57.42°	FOV horizontal 57.42°	FOV horizontal 50.26°
FOV vertical 45.50°	FOV vertical 34.23°	FOV vertical 29.38°

Table 1: Overview of available image data

2.2.2.3 Focus, exposure & white balance modes

The same framework also provides accesses to the camera's focus, exposure and white balance modes. The mode options include FocusLocked, AutoFocus and ContinuousAutoFocus and ExposureLoked, AutoExpose, and ContinuousAutoExposure while

white balance options are reduced to WhiteBalanceLocked and ContinuousAutoWhiteBalance (Apple Developer Library 2013:55).

In auto-mode, these features work very fast. But as they seem to adapt primarily to the centre section of the image, the output's look easily "jumps" from one extreme to the other when the camera is moved.

The auto-focus tends to change rapidly between far and near objects – often even without moving the camera. While snapping to its new aim, the video blurs.

The auto-exposure adapts fast to very bright sections in the visible field, easily clipping the rest of the image. Dark sections are rather neglected.

The auto white balance fully focuses on the light temperature in the centre of the image instead of compromising between different ones present in the view. Therefore, objects of one specific

light temperature are well balanced but the rest of the visible field appears strongly coloured and thereby alienated.

2.2.2.4 Video quality

When assessing the overall image quality of the video stream, the camera definitely delivers crisp images of relatively high dynamic range. Only the lower mid-tone range, particularly grey shades, shows quite noticeable noise. Blacks, though, remain unaffected by that and highlights surely tend to clip but not to bloom.

The video image is subject to motion blur: The more distant the focused objects, the more they blur if the camera is moved.

2.2.2.5 Conclusion

Reviewing these characteristics and options for the design of the MRS, the following points should be considered:

- If the final MRS design should rely on the analysis of the camera's video stream, the raw option would offer more than double level of detail compared to the resolution of Full HD – provided the raw data can be tapped in Unity. But with triple-fold LOD comes triple-fold data volume to be processed with every frame, which might be expensive on the overall system's performance. Should the final MRS indeed rely on the stream, it would have to be tested in the prototype if the higher LOD leads to results that justify their additional data load.
- As regards the available FOVs, the non-stabilised video stream would offer a more human-perception like image source, at least as regards the human's central field of vision. However, the missing stabilisation feature might lead to shakier and consequently blurrier results. This might diminish the output quality – and thereby the desired illusion! (the real-world view blurs – the CGI layer does not) – more than a slightly more narrowed field of view. As a trade-off, the stabilised video stream is the option of choice.
- The overall video quality is basically only negatively affected by the noise occurring in mid-tones. This should be considered as a trade-off costs of having a small-sensor camera available in a mobile phone and will state only small imperfections in the final output result.
- For avoiding further blurring and achieving a stable image look, the focus/exposure/white balance modes should be locked

2.2.3 iPhone 5 display

The display measures 4 inch diagonally at an aspect ratio of 16:9 and offers a resolution of 1136 x 640 pixel at 326 ppi. Further specifications are the maximum brightness of 500 cd/m^2 and contrast ratio of 800:1 (Apple 2013b).

Its high resolution and contrast ratio allow a well-detailed output quality.

The quite small display covers only a small section on the users field of view so that he always sees the unmediated reality as well. Also the phones layout puts a clear "frame" to his field of vision. These given factors clearly diminish the illusion of being fully immersed into the MRE. The display rather serves as a viewing window through which the user can see a "hidden world".

Unlike a (translucent) head mounted display that could provide a more immersive effect, smartphones provide the advantage that they are wide spread and do not rely on restricting cabling.

2.2.4 Possible site restrictions

Site operators and visitors alike surely would prefer the exhibition spaces to be kept as unaltered as possible.

The MRS should therefore not require external set-ups that might appear intrusive or even distractive to the visitors, e.g. like well-visible markers or sensors. Should it turn out, though, that such set-ups cannot be avoided, they should be few and fit-in well into the room's ambience.

Sites of cultural heritage often exhibit sensitive artifacts, for which special technical measures have to be taken in order to keep them accessible for the public but also to further preserve them for future generations. Particularly paintings are subject to discoloration over time if exposed to ultraviolet or infrared light (Stirton 2013). Such objects are therefore exhibited under special lighting that has been filtered off these two spectrum areas. This is also the reason why artefacts are usually not be photographed using flash light. Some exhibition rooms even require overall dimmed light.

When designing the MRS, it should therefore be taken into consideration that the system might also be need to work with lower-contrast camera images and that external set-ups should not require UV / IR radiation (e.g. certain sensor types).

2.3 SUMMARY OF TECHNICAL FRAME CONDITIONS

The technical frame conditions resulting from the MRS's HCI design concept and resource constraints can be summarised as follows:

I) MRS system general

- For ensuring the targeted output update quality (between 15 and 24 Hz), the overall system's end-to-end latency has to be at average 42 ms/update below 66 ms/update as an absolute maximum …
- … while overall CPU and memory load must lie within the iPhone mobile computing capabilities to ensure the required performance

II) Hardware

- The visitor handheld should not be cable bound or otherwise restrictive to use
- The iPhone camera's focus, exposure and white balance modes will have to be locked for ensuring a stable video look

III) Tracking system

- A sufficient registraton result can only be achieved with high accuracy and frame synchonous tracking data
- It needs to provide continuous tracking data of 24 Hz absolute minimum stable real-time for ensuring temporal continuous proprioception.
- The tracking system should cover whole room or: the tracking system should be able to retrieve reliable data at any position of the room so that the user can explore it at maximum freedom
- It should be robust enough to allow the user to move the handheld at a comfortable speed
- If external set-ups cannot be avoided, these should be non-intrusive and be able to work without the use of UV/IR light, as these might be conditions of cultural heritage sites.

- For avoiding external set-ups in the first place, the tracking system should preferably be implemented in-phone. If such a tracking system relies on the analysis of the camera video stream, it might have the able to deal with the lower contrast of dimmly lit rooms.
- This might also make necessary to provide a higher resolution source, i.e. the iPhone raw video stream.

IV) MRE set-up & design

- The digitalised real-room replica should include all relevant objects regarding occlusion and population interaction. The reproduction should be of a sufficient accuracy and vertices count. The location of the single objects to each other also has to be reproduced precisely.
- The design of the virtual props should be adequate but also cost-efficient as regards vertices count, texturing, shading and lighting
- The depiction of the virtual props will have to correspond their real-world environment and to the video quality of the phone's camera.
- The virtual camera and moveable virtual objects will be equipped with basic game physics (Psi/basic physical interaction-reactions)) and sound.
- The virtual camera characteristics need to be matched accurately to those of the iPhone camera
- The output of the composited video has to be at a real-time update rate of minimum 15 and average 24.

V) Usability

- The GUI design, if even necessary, should be kept minimal
- A main menu with use instructions and restart button should be provided
- The error handling should work as self-sufficiently as possible to not disturb the user
- If recovery instructions are necessary, they should be and non-technical

2.4 ABSOLUTE CONSTRAINTS

The analysis has also shown that the iPhone's characteristics do impose some absolute limitations to the MRS design resulting in the fact that the targeted user illusion of maximum immersion cannot be achieved to the full:

According to the PI conditions, the MRS should facilitate that the MRE can be (visually) perceived by a user "in the same way as perceiving the real world"(Slater 2009:4 sqq.).

It has been therefore been stated in the conditions of PI and the subsequent HCI design concept that the MRE's field of view should match at least the human's central field of vision.

Unfortunately, the iPhone's camera does not even meet this minimum requirement: As a trade-off solution for more stable images, the camera's narrowest FOV of 50.26° h / 29.38 v applied. Compared to the human field of vision of approximately 50-60° h / 55° v, the constraints become clear: The MRS only provides a noticeably narrow view on the MRE.

It has also been stated that extent of immersion also largely depends on how much the MRS's display can cover the user's field of vision: The less the user can see of unmediated reality while exploring the MRE, the higher the degree of immersion. But as the iPhone's display is rather

small and intended to be hold at reading distance, much of unmediated reality intrudes the users vision and its surrounding frame even puts a strongly visible border between the two realities.

Concluding, instead of a maximally immersing experience during which the user feels “surrounded” by the MRE, the phone is therefore rather a “viewing window” with which the user can peek into “a hidden world.” (Naimark 1991:1).

3 REALISATION POSSIBILITIES

This section will examine possible implementation approaches and evaluate them for their suitability for the MRS in accordance with the given frame conditions. The section will conclude with a comprehensive solution suggestion.

3.1 POSSIBILITIES FOR CREATING THE VIRTUAL WORLD

The “virtual world” part of the MRE will mainly be implemented with the real-time 3D game engine Unity (by Unity Technologies ApS), version 4.3.2f1. It provides all tools and functions necessary for designing the MRE:

“The game engine is the core component of a game or simulation environment, marshalling all the assets and managing their availability, rendering, and behaviour. Game engines usually also provide game AI, collision detection, camera placement, lighting, shadowing and shading, game physics, heads-up displays, and other features”(Iuppa 2006:188). Moreover, Unity is based on Mono and supports scripting in C#, Boo and JavaScript.

The design of the MRE is done in the following stages: First, the necessary real-world components have to be reproduced in Unity, including the room itself, its lighting conditions as well as the properties of the real world camera. Then, scenario-specific virtual props and characters have to be designed and staged in the virtual room and added interactive behaviours. As a last step, the scene has to be optimized for efficient rendering and prepared for the composite output.

The following section will point out what has to be considered during the design process so that the related frame conditions are met.

3.1.1 Reproducing the real-world environment

3.1.1.1 Retrieving furniture and other objects

Pieces of furniture and objects with flat surface can be measured manually. Moreover, of many modern products detailed 3D data is available nowadays.

For filigree and uneven objects, 3D scanning is a suitable option:

Autodesk’s 123d catch uses photogrammetry to create 3D models from a minimum of three photographs of an object. The photos have to be uploaded to an Autodesk server for processing. The tool’s reproduction accuracy of 1:600 (Chandler and Fryer 2011) should be sufficient for use in the MRE.

For testing its scanning quality, 21 photos from various angles of a test object were used to reproduce its 3D model. The result shows that the mesh was reproduced with accurate proportions but considerable flaws.

Fig. 5: 123d catch texture result

Fig. 6: 123d catch mesh result

Kinect Fusion as part of the Kinect for Windows SDK 1.8 offers 3D scanning and model creation: Based on the depth map produced by the Kinect hardware's IR projector and camera, "the Kinect Fusion system reconstructs a single dense surface model with smooth surfaces by integrating the depth data from Kinect over time from multiple viewpoints […] and this resultant point cloud can be shaded for a rendered visible image of the 3D reconstruction volume." (Microsoft Kinect 2014). The Kinect's maximum resolution is 768 voxels (scanning volume units), the dimensions of which are scalable. This means that either a small volume can be scanned with a dense, small-sized voxel grid and hence high resolution, while scanning a large volume requires that also the voxel grid is scaled up accordingly, resulting in a lower detail resolution.

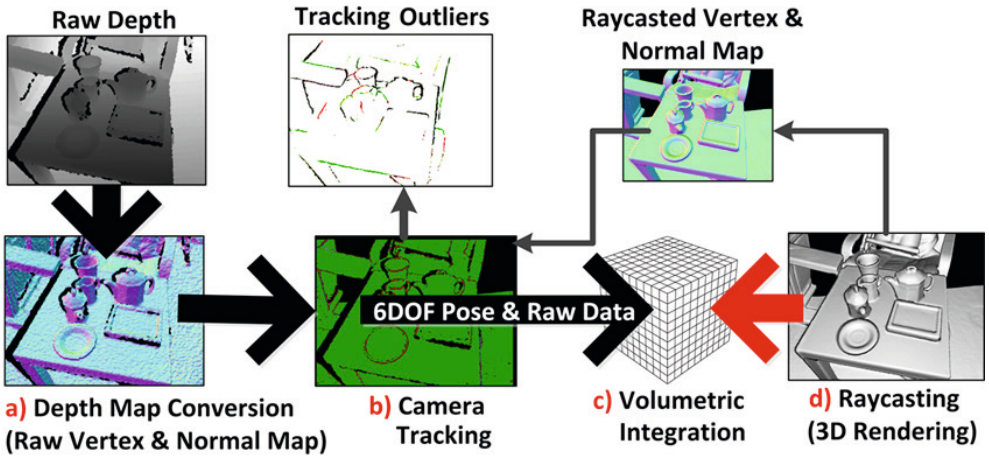

Fig. 7: Kinect Fusion Pipeline (Microsoft Kinect 2014)

The scanning quality of Kinect Fusion was tested by using the "Kinect Fusion Explorer-WPF" implementation of the Kinect developer kit. It allows to manually adjust the voxel grid density per metre and hence the dimensions of the scanning volume and resulting mesh quality. For the test scan, these parameters were set to fit the dimensions of the test object (0.18 m^3). The scan result shows a highly accurate mesh of ca. 840,000 vertices:

Fig. 8: KinectFusion scanning mesh result

The scanned object can be exported to the file format OBJ, which is also supported by Unity.

Concluding, Kinect Fusion provides suffciently accurate reproduction results and is therefore a suitable solution for retrieving small to mid-sized room objects.

3.1.1.2 Retrieving the general architecture

The general architecture of the room, i.e. walls, stairs, doors with their dimensions and their positioning to each other, can be retrieved by manual measurements or (for rather modern buildings) from architectural plans. It should be double-checked, though, if these truly correspond to the actual construction.

However, particularly ancient sites of cultural heritage often feature uneven or highly detailed architecture, e.g. uneven walls, floors and filigree staircases, and can therefore barely be reproduced by manual means.

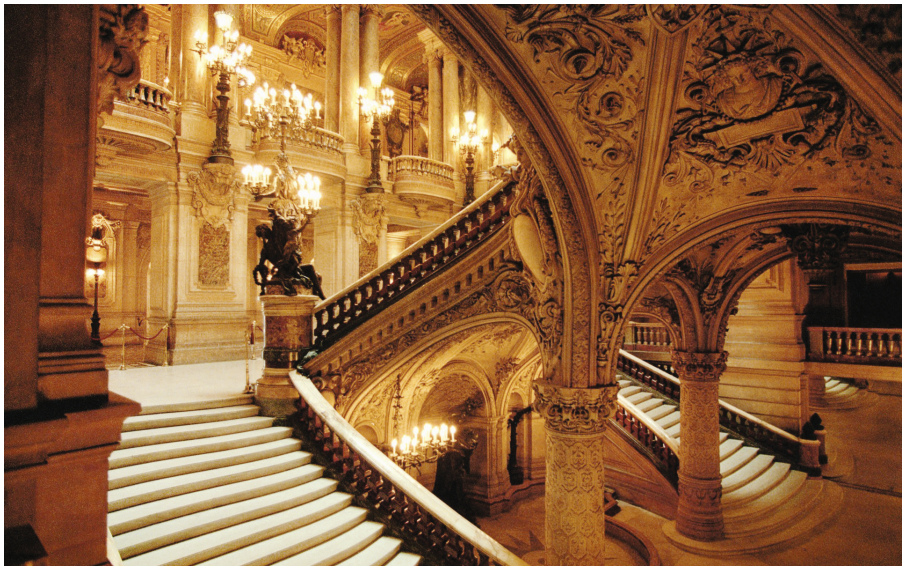

Fig. 9: Example for a very filigree, uneven site environment ("Stairs in the Castle", n.d.).

These could be digitalised using large-volume 3D scanning technologies:

The software Skanect leverages the 3D scanning possibilities of the Kinect sensor in similar ways as KinectFusion but is capable of covering far larger scanning volumes of up to 12 x 12 x 12 metres. This scanning method is an interesting approach for realising the room's retrieval for being usable with a widely distributed sensor and standard laptop computing capabilities. The general feasibility of a full room scan with Skanect was tested with a Kinect sensor that was placed in the middle of a suitable test room and rotated around its own y-axis by 360°. The result shows that the Skanect is indeed capable of reconstructing large volume environments but as its integration process is subject to drift, the reconstruction errors increased over the course of the scanning circle. The reproduced test mesh therefore shows e.g. offsets between the floor levels of the beginning and the end of the scanning process. Furthermore, the resolution of the mesh turned out to be too low for being used as an adequate occlusion model.

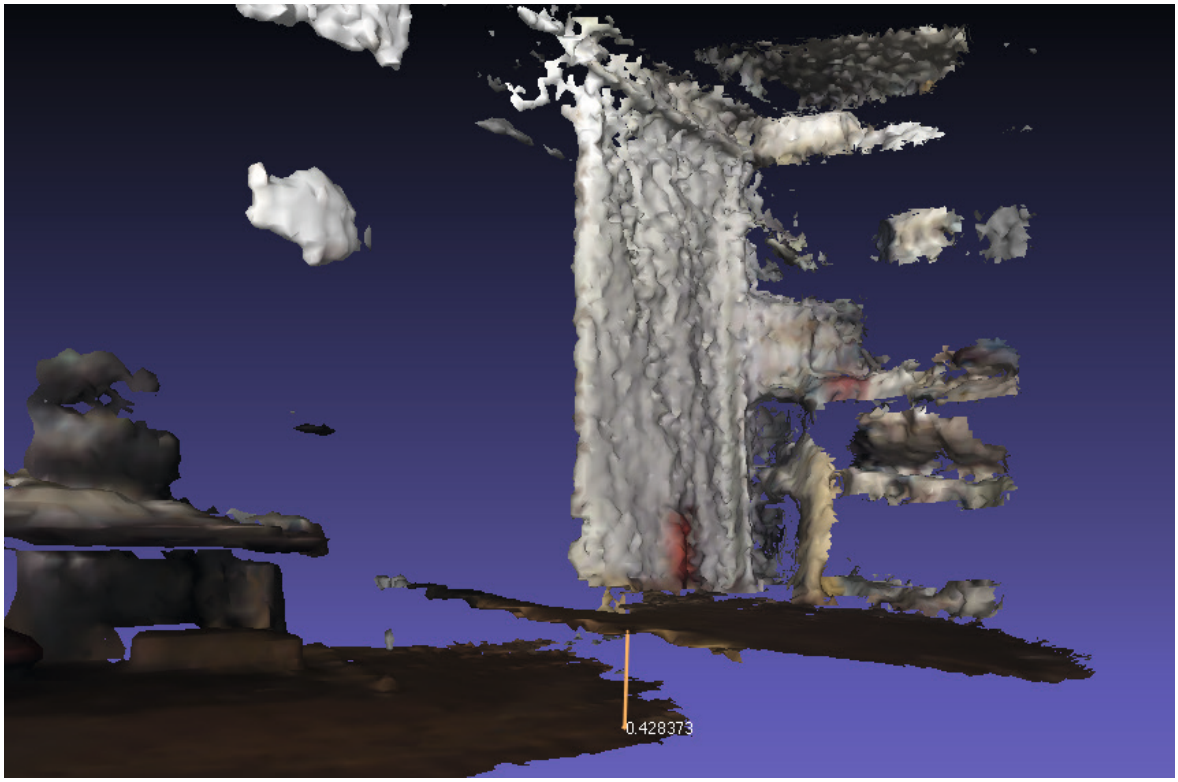

Fig. 10: The Skanect test scan result shows ca. 43 cm floor offset and low mesh resolution

Hence, a more sophisticated system would be necessary:

British organisation CyArk has been conducting the digitalisation of numerous major sites of cultural heritage, among which Pompeii and the Maya city of Tikal, using long range terrestrial laser scanners (CyArk 2013). These scanners measure the laser beams' time of flight as well as their vertical and horizontal return angles. From these data x/y/z point coordinates of 2-3 mm accuracy are then calculated. They are stored in large-scale point clouds (Kacyra 2011). Big data management software is then used to "align multiple clouds from different vantage points into a single coherent scene" (Soulard and Bogle 2011) and process them into high resolution 3D data (Frei, Kung, and Bukowski 2005:2) that can then be scaled down depending on its use purposes.

3.1.1.3 Arrangement in Unity & depth-mask shading

Once the 3D models of the room's architecture and props have been retrieved successfully, they are arranged in a Unity scene according to their real-world dimensions and positions.

In order to keep the models' total processing expenses at an economic-efficient level, it might be necessary to downsize their resolution accordingly. As stated in the basic requirements, though, the precision of the occlusion model is a crucial factor to the accuracy of the later registration. The resolution reduction should hence not compromise the models' accuracy.

As the room replica serves as an occlusion model, all its elements are added a depth mask shader. In this shader, the parameter "Lighting" is turned off, so that neither the object itself nor anything that it occludes will be drawn in the final output but masked out.

Fig. 11: Occluded object

```
"DepthMask" {

bShader {

  Tags {"Queue" = "Geometry-

  Lighting Off

  ZTest LEqual
  ZWrite On

  ColorMask 0
  Pass {}
```

Fig. 12: Depth mask shader

Fig. 13: Depth mask applied

3.1.1.4 Reproducing the lighting situation

For reproducing the lighting situation of the room, Unity offers the following options: Directional lights "are placed infinitely far away and affect everything in the scene, like the sun" (Unity Documentation 2013a). A directional light is therefore set and adjusted to mimic the sunlight falling through the windows of the room.

Point lights shine "equally in all directions from its location, affecting all objects within its range" (Unity Documentation 2013a) and will therefore be used as the room's lamp lights.

The scene's ambient light will be used to substitute the light that is reflected by walls and room objects.

The intensity of the light sources has to be adjusted in a way that the overall contrasts and total brightness of the scene match the iPhone camera's image.

3.1.1.5 Reproducing the iPhone camera's field of view & camera set-up

A camera object, which represents the real world iPhone camera, is added to the scene. In its component settings, the camera can be adjusted to match the iPhone's characteristics by setting the FOV parameter (vertical) to 29.38° and the viewport aspect ratio to 16:9.

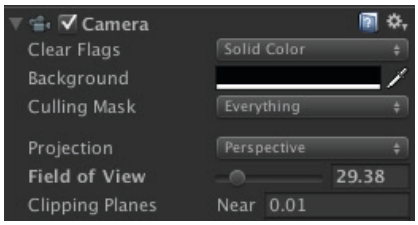

Fig. 14: Unity camera component

For the further camera setup, the devices focus, exposure and white balance modes need to be locked. This cannot be done in Unity but in its deployed Xcode project files: The related ModeLocked setter-methods of the AVCaptureDevice class / AVFoundation framework have to be set as shown in the figure below:

```
void LockCam(){

    NSArray *devices = [AVCaptureDevice devicesWithMediaType:AVMediaTypeVideo];

            for (AVCaptureDevice *device in devices) {
        if ([device lockForConfiguration:nil]) {
            [device setWhiteBalanceMode:AVCaptureWhiteBalanceModeLocked];
            [device setExposureMode:AVCaptureExposureModeLocked];
            [device unlockForConfiguration];}

}}
```

Fig. 15: Locking of the iPhone camera modes

3.1.2 Designing the virtual world

3.1.2.1 Design considerations for virtual objects

In Unity, virtual objects can either be based on internally generated primitive shapes, e.g. cubes, spheres, etc. or externally created meshes with up to 50,000 vertices imported via FBX or OBJ files. The meshes can be added materials, defining their texture and shading.

Virtual characters require rigged meshes in order to be added animated behaviour.

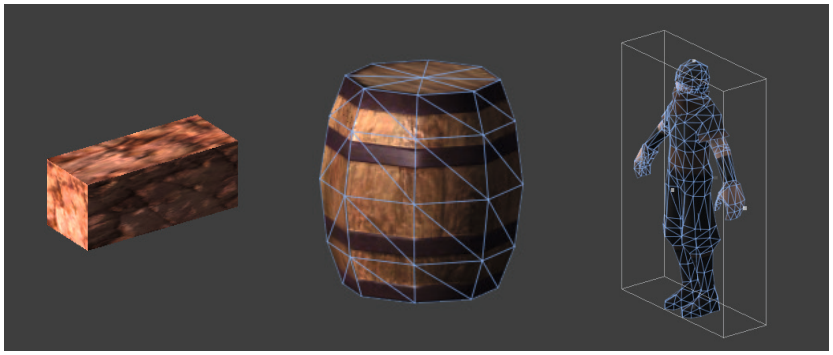

Fig. 16: Examples for virtual objects with varying mesh complexity

Fig. 17: Example for rigged character mesh

As stated in the frame conditions, the virtual objects of the MRE should be designed in a cost-efficient way in order to not affect the system's real time performance, i.e. being recognisable as what they represent while being suitable for real time rendering.

Imported meshes should therefore have priorly been reduced to a sufficient LOD or vertices count, respectively, and textures should be of a reasonable resolution. Applying Unity's internal compression can further optimise meshes and textures alike.

Regarding shaders, Unity offers a set of options specially geared for rendering on mobile devices. Most suitable for the character's design are either the diffused or specular bumped shader. Bumped texture mapping is used to "create the illusion of surface relief (elevations and depressions) on an otherwise flat object" (Maya User's Guide 2013) while not altering its actual mesh geometry. They can therefore be used to enrich an object with fine-grained details at much less expenses than using the equivalent mesh structures.

Fig. 18: Application of bump map to a corresponding texture

3.1.2.2 Adding basic interactive behaviour

As analysed from the requirements of achieving PsI, the user should be incorporated into the MRE as a physical body. The virtual camera should therefore not be able to move "through" virtual objects but props should move and characters should step aside if they collide with the virtual camera.

I) Preparing virtual objects as rigid bodies with sound effects

For letting the virtual props behave like rigid physical objects, they have to be applied game physics, which is provided by Unity's rigidbody and collider components.

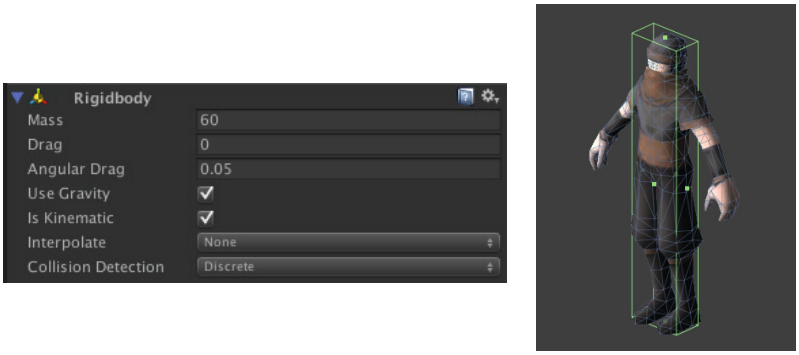

Fig. 19: Rigidbody component and collider visualisation

The rigidbody component facilitates the objects physical behaviour by making it responsive to the virtual worlds forces according to its defined mass and drag. The applied collider defines the boundary area of an object. If the prepared object is then exposed to a force, e.g. exerted by another moving object, it will move aside accordingly.

This can further be underlined by adding corresponding sound effects. Therefore, the object has to be added a sound source component equipped with a sound file. In a related script, the sound can be set up to be played once their collider detects a collision with another object's collider.

```
function OnCollisionEnter(collision : Collision) {
     if (collision)
          audio.Play();
}
```

Fig. 20: Event handler function for collider

Also the virtual camera has to be added a collider and rigidbody component for making it a rigid object. But as stated in the prior analysis, the user's physical representation in the MRE

should be able to exert force onto its objects – but the MRE's objects cannot exert force on the real-world user. Therefore, the camera's rigidbody component has to be set to "kinematic" so that it can exert but not receive force (Unity Documentation 2013b).

II) Letting characters walk through the environment while avoiding the user

If the virtual population is supposed to move around the MRE, they require a certain knowledge where they can move or how to avoid the room's obstacles, respectively.

For implementing this, Unity's pathfinding possibilities can be used: A navigation mesh is baked from the static room model, showing all walkable areas. The characters are added a Nav Mesh Agent component, which implements the pathfinding based on the provided navigation mesh. On runtime, the component calculates a path between the current position of its game object (here: the character) and the position of a provided target object. Once the path has been established, the character is translated towards it. The velocity and acceleration of the translation can be set in the Nav Mesh Agent component's parameters.

As a result, the virtual characters will avoid the room's static obstacles or walk around them, respectively, for reaching their target position.

For letting virtual characters make way for the user,a dynamic object, the virtual camera is equipped with a Nav Mesh Obstacle component. By using this component, an adjustable area around its game object is carved out from the baked navigation mesh. This area is updated dynamically as the game object moves through the scene. The character's navigation path calculation is updated accordingly in order to avoid the dynamic obstacle area.

The characters will therefore walk around the virtual camera if it stands in their way. If the characters have already arrived at their target object, they will move away from it once the user's obstacle area enters their Nav Mesh collider. Once the target object is unblocked again, the characters will navigate back to it.

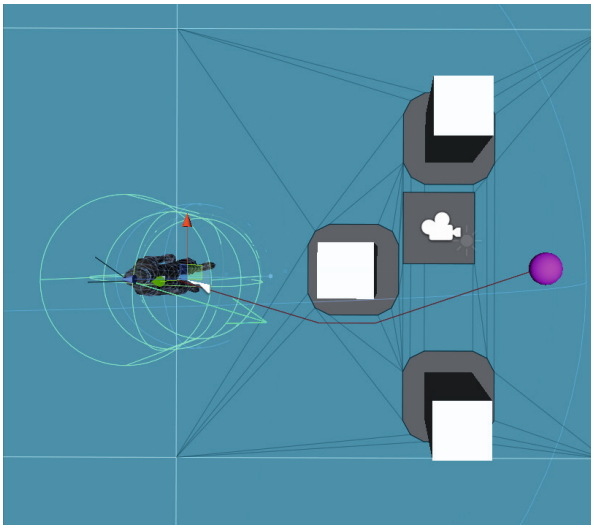

Fig. 21: Pathfinding for obstacle position 1

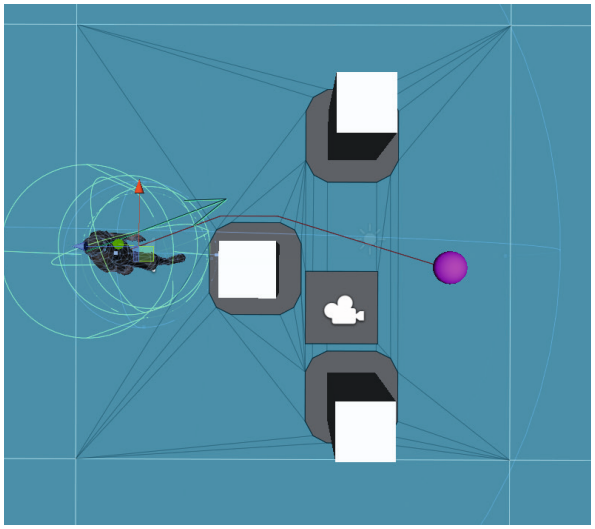

Fig. 22: Pathfinding for obstacle position 2

3.1.2.3 Staging and interaction possibilities for storytelling

The readily prepared characters can now be used for implementing the storytelling concept of the specific scenario.

The possibilities herefore reach between purely time-linear dramaturgy and highly dynamic, user-dependent storytelling.

Time-linear storytelling implies that the actions of virtual story-characters follow a timed, consecutive order. In Unity, this can be implemented by binding the animations of characters to certain time events, i.e. that a specific animation is activated once a certain point in time has been reached.

```
if (Time.time==20){
    animator.setBoolean("Animation_Waving", true);
    }
```

Fig. 23: Very basic example for binding animations to time points

In time-linear storytelling, the user is thus not involved in the unfolding of the events. He can explore the MRE and watch the plot unfolding from a perspective of his choice.

Dynamic, user-dependent storytelling can be realised by applying "the interactive mechanisms used in games" (Champion 2002:1). The user takes the role of a fictional character of the scenario and is given an aim to pursue as an incentive to engage with the virtual object. Possible game goals for an MRE scenario could e.g. be to reach a certain point, find a hidden object, catch another character, etc. The achievement of these exemplary goals can all be detected by using collider zones as event listeners: Once the player's collider enters the collider zone of the place to reach or the character to catch, the event handler functions OnCollisionEnter / Exit or OnTriggerEnter / Exit of the Unity collection are triggered.

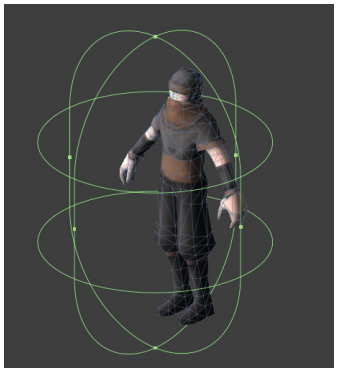

Fig. 24: Example equipping virtual objects with trigger collider zones

This goal becomes meaningful, a challenge, if obstacles have to be overcome to reach it (Iuppa 2006:126). Apart from setting a ticking clock or hiding the target object under virtual props, the virtual population can be given a driving role here: If the user e.g. has to find an object or solve a puzzle, he can approach the virtual characters for telling him the necessary hints and instructions. Again, this could be implemented by attaching trigger zones to the characters, the event listeners of which prompt them to rotate towards the user and play a corresponding sound file.

The characters can as well be set up as direct obstacles by using the NavMeshAgent component: The characters could for one chase the user to block his way or attack him or – if it's the user to chase them – run away. There are thus plenty of possibilities to actively engage the user into the shaping of the scenario's plot.

The design of a scenario's storytelling is not necessarily bound to using either time-linear or dynamic storytelling, but can be merged along a continuum between both extremes.

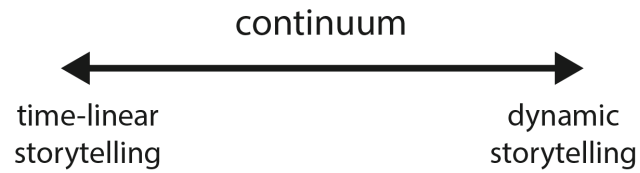

Fig. 25: The continuum ranging between the time-linear and dynamic storytelling

3.1.2.4 Preparing the composite output

The final output result consists of the virtual camera's masked-out viewport output which is superimposed on top of the iPhone camera's "real-world" video stream. For compositing the streams, both have to be retrieved in layerable formats:

The virtual camera's viewport output can be stored in a render texture object.

The iPhone camera's video stream is addressed as a WebCamDevice object. For the retrieved video basically being a 2D (moving) image, it can be stored as a texture. Unity provides the WebCamTexture class which is specifically geared for moving images. Therefore, a WebCamTexture object is created, specifying the desired device from which to retrieve the live video input.

```
private WebCamTexture webCameraTexture;

void Start () {

    webCameraTexture = new WebCamTexture(WebCamTexture.devices[0].name, Screen.width,Screen.height,25);
```

Fig. 26: Retrieval of the iPhone's video stream

With both imagery sources now being two dimensional images stored in texture objects, they can be layered on top of each other by using GUI layers. These are components attached to a camera-object for rendering 2D imagery on top of the camera's viewport. Therefore, two GUI texture objects are created, each one being assigned one of the prepared imagery source textures. The layers' hierarchy is defined by the GUI textures' z-position. Thus, the render texture is assigned the higher z-value as it has to be displayed on top of the live video.

The main camera is assigned a GUI-layer render component so that it outputs the two layers.

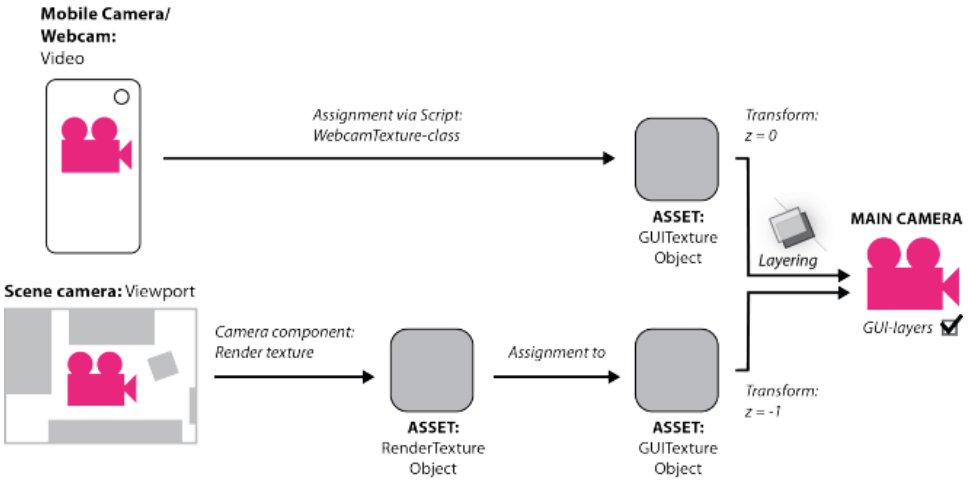

Fig. 27: Retrieval and compositing workflow scheme

3.1.2.5 GUI screens

For ensuring usability, the medium should offer 2D GUI screens that provide the user recovery instructions in case system breaks as well as a help menu with a restart and resume option.

Unity provides the UnityGUI collection that can be implemented within the related OnGUI function. The GUI class provides methods for drawing standard elements, including boxes, labels, buttons, etc. that are already equipped with click and touch event-handlers. The drawing of various GUI layouts can be set up to be triggered by certain events.

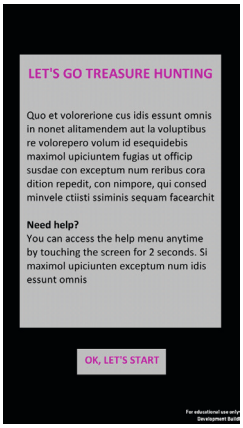

Fig. 28: GUI start screen example

The display of a user instruction screen could be made dependent form a related boolean variable that switches its value once the system assesses that e.g. tracking system broke. In this case, a GUI screen with instructions is drawn until the system has recovered before the depending boolean is switched back to its original value.

```
void OnGUI () {
if (GUI_notracking_info ==true){

    //content
    GUI.Box(new Rect(Screen.width*0.07f,Screen.height*0.7f,Screen.width*0.85f,Screen.height*0.24f),"<size=37>OOPS, NO IMAGE IN VIEW FIELD</size>");
    GUI.Label(new Rect(Screen.width*0.1f,Screen.height*0.835f,Screen.width*0.8f,Screen.height*0.13f),"<size=30>Point your camera at one of the poste
}
```

Fig. 29: Screen design scripting in the OnGUI function

A help menu, on the other hand could be triggered by a longer touch event by the user, switching the GUI's drawing boolean. The resume button could be set up to switch the same boolean again while the restart button prompts the re-loading of the level.

```
void Update () {

//---- iPhone Touch more than 2 sec = GUI Menu ----
    for (int i=0; i < Input.touchCount; i++)
    {
        if (Input.GetTouch(i).phase == TouchPhase.Began)
        {print ("touch");
            startTime = Time.time;

        }
        realTime = Time.time-startTime;

        if (Input.GetTouch(i).phase==TouchPhase.Ended){

    }
        if (realTime>2){GUI_touchevent=true;}


    }
```

Fig. 30: Touch condition for prompting the help menu

3.1.2.6 Final rendering optimisation

Apart from using reasonable mesh and texture resolutions, the graphics' performance can further be aided by reducing the necessary draw calls of the scene: "To draw an object on the screen, the engine has to issue a draw call to the graphics API (e.g. OpenGL or Direct3D). The graphics API does significant work for every draw call, causing performance overhead on the CPU side" (Unity Documentation 2013c).

I) Batching

A considerable reduction of draw calls can be achieved by "batching": Unity can combine all those elements in one single draw call that use the same material.

Therefore, the MRE should use as few materials as possible that are shared between its virtual elements. Different textures can be used as the same material by combining them in one large texture, also known as texture atlasing (Unity Documentation 2013c).

II) Occlusion culling

Unity automatically applies frustum culling, which means that those virtual elements outside of the camera's current field of view are not rendered. In addition to this, also occlusion culling can be applied so that also those objects that are in the camera's field of view but hidden by other object are not rendered as well, saving also their draw calls (Wiebe 2011:40).

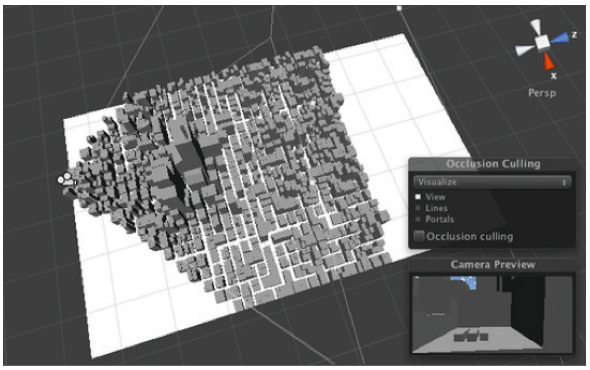

Fig. 31: Scene with frustum culling (Unity documentation 2013e)

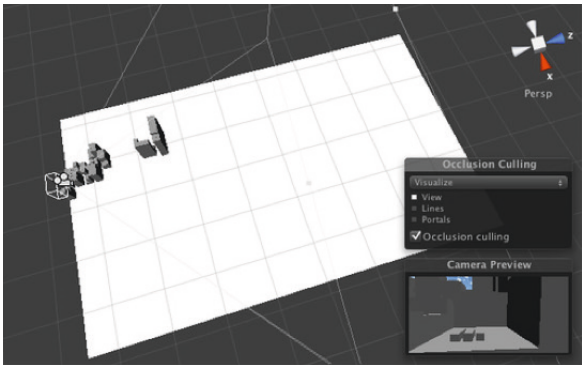

Fig. 32: Scene with additional occlusion culling (Unity documentation 2013e)

III) Lightmapping using light probes

Using dynamic lighting, which is calculated at real time, is cost-expensive. Therefore, the "baking" of the lighting situation into the texture of an object, known as lightmapping, is common practice to reduce render expenses considerably. Standard lightmapping, though, only applies for static objects (Wiebe 2011:16) – all of which the MRE's virtual elements are not. Unity therefore offers the use of light probe, which allow light mapping for dynamic objects: A grid of probes, represented by small spheres, is spanned over the room. On baking, the probes store a sample of the light surrounding them. "The idea is that the lighting is sampled at strategic points in the scene, denoted by the positions of the probes. The lighting [of a virtual object] at any position can then be approximated by interpolating between the samples taken by the nearest probes" (Unity Documentation 2013d).

3.1.2.7 Performance monitoring

The so-far prepared project is deployed on the target output device, in this case the iPhone 5, for performance testing.

On runtime, the graphics performance can be monitored using Unity's built-in profiler, which is also available in Xcode's debugging console. Monitored statistics include the overall achieved update rate and its various dependencies, e.g. main thread and rendering latencies, required draw calls and memory usage.

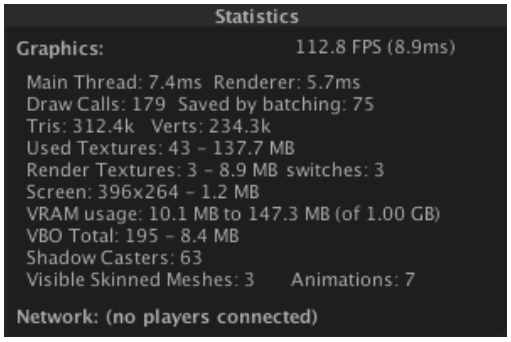

Fig. 33: Statistics screen

Should the statistics show that the targeted graphics update rate cannot be achieved by the capacities of the target device, the overall scene's content would have to be further reduced until an acceptable update rate is reached.

3.2 TRACKING POSSIBILITIES

It has been lined out in the section “General technical principle & system architecture”, that the matching of the virtual with the real camera will be achieved via a tracking system.

According to the related frame conditions, a suitable tracking system needs to provide highly accurate tracking data, full and permanent room coverage, a latency short enough to not affect the overall system latency to more than the targeted 42 ms /update and should preferably not rely on intrusive external set ups, hence be implemented in-phone.

This section will analyse various available tracking techniques and evaluate them according to the stated MRS requirements.

3.2.1 Passive tracking techniques

Passive tracking techniques “are mounted on the user” (Strand 2008:25) and can produce tracking data while the user moves their implementing device. “They are typically visual trackers which use features in the surroundings to pin-point the users location and orientation” (Strand 2008:25). Visual tracking uses computer vision (CV) methods which analyse video data for specific feature types and, once recognised, track their movements from frame to frame (Yilmaz, Javed, and Shah 2006:2). In mediated reality applications, the retrieved data is used to calculate the corresponding virtual camera pose estimate in accordance to the tracked object (Lima et al. 2010:1, 3).

Using passive tracking techniques in the MRS might state a suitable solution approach: They can be implemented in-phone and hence do not rely on external set-ups. As the same video frame that is used for the tracking data analysis is also the one onto which the correspondingly calculated virtual video frame is superimposed, the resulting registration quality should consequently be sufficient.

It has to be considered, though, that an in-phone implementation of a tracking system also implies a higher processing load on the phone’s resources and therefore affects the latency of the overall system. Another decisive criteria point is therefore if these suffice with computing capacities of the iPhone 5 while also leaving a certain share of resources to other system components, e.g. the graphics rendering.

In the following, common feature based visual tracking techniques will be analysed. For those techniques most fitting the requirements of the MRS, possible implementations will be evaluated.

3.2.1.1 Feature based visual tracking techniques

Available techniques work marker based or markerless: Marker based tracking relies on placing extra markers in the environment. But as they would alter the original environment, they are considered intrusive (Lima et al. 2010:1) and are thus to be avoided.

Markerless techniques, however, use the environment’s natural features for tracking. Also they subclass into two major types:

Those based on Structure-from-Motion (SfM) create tracking data while moving the camera in an environment. Consequently, these techniques do not require any prior knowledge of the environment to track. But this results rather in a general mapping of an environment than the localisation of a camera within a known room. These techniques are therefore not applicable for the MRS.

Model based techniques recognise known characteristics of an environment for then matching the camera pose estimate. When it comes to the concrete features types to be tracked, these techniques either rely on detected pixel flows during movements, edges or textures.

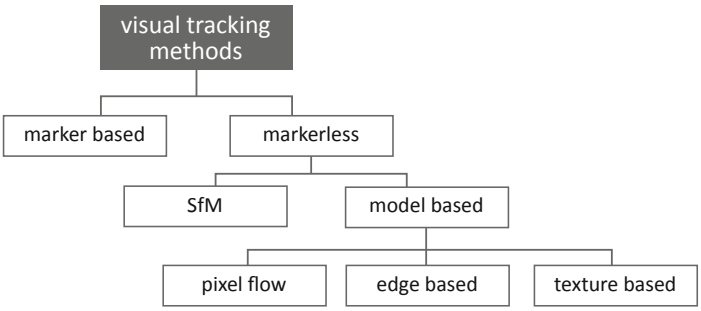

Fig. 34: Flowchart visual tracking methods (in accordance with Lima et al. 2010)

Pixel flow methods obtain tracking data by analysing the motion of pixel patterns over the camera sensor during movement (Pankratz 2009:31). However, this would require constant lighting conditions and the absence of any other moving objects within the field of view (Lima et al. 2010:5). This is not true for sites of cultural heritage due to other visitors being present in the room. This feature type is therefore not applicable.

Edge based tracking can be achieved by matching a provided 3D wireframe model against noticeable gradient edges in source images (Lima et al. 2010:5; Wuest, Vial, and Stricker 2005). This technique is therefore suitable for tracking polygonal and strongly contoured objects (Lima et al. 2010:5; Yilmaz, Javed, and Shah 2006:6). Contoured architectural components and furniture in the MRE could therefore serve as possible targets. Moreover, their wireframes would already be provided and hence not add further to the offline data load.

A common egde based trackin technique is the point sampling method: As preparation data, control points are sampled along the edges of the wireframe. Then an initial pose of the wireframe object has to be determined from which the objects initially visible edges are detected. During online phase, the remaining sampling points are compared to the images' contrast edges. The pre-determination of an initial pose consequently implies, however, that the tracking process has to initialise from a certain starting pose of the object to track.

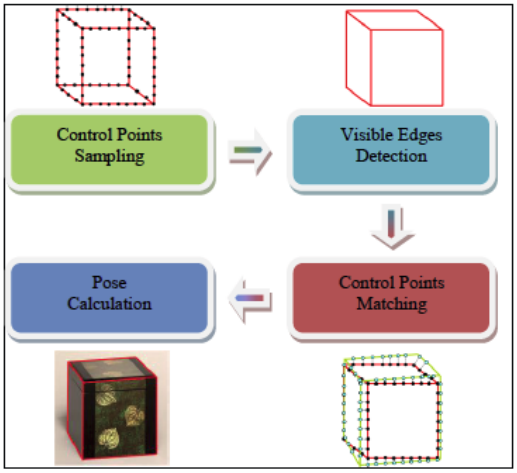

Fig. 35: PS technique workflow (Lima et al. 2010:6)

For the MRS, this would mean that users would have to initialise the overall MRE by pointing the handheld device at an edgy object as kind of a calibration point. If tracking breaks, the system would not be able to recover automatically. Either the user would have to reinitialise the system by going back to the calibration point or the system would require data from a second tracking system. The first case cannot be compromised with the stated usability requirements. Concluding, edge based tracking cannot not be used as the only tracking system in the MRS. But it should be considered to use it as a secondary system in order to include contoured real-world objects as possible tracking targets.

Texture based techniques as the keypoint based method compare known texture characteristics to current video frames. In a prior training phase, the texture feature database is prepared: Images of each face of an object and their corresponding camera pose serve as the basic resources. On these images, the texture of the face should be well visible and undistorted. They should therefore be photographed from an orthogonal viewpoint.

The textures are analysed for unique features that "are invariant to viewpoint, scale and illumination" (Lima et al. 2010:10). These are strong, pointed contrasts in the texture's structure which remain indifferently well visible also if lighting conditions change or if they are seen from far, near or angled viewpoints. With reference to the MRE, suitable tracking targets would therefore have to be flat high-contrast structured surfaces with unique details, e.g. strong wood textures, paintings, prints, photographs, etc. These extracted features and their position in the source image, related camera pose and their resulting 3D position describe a "keypoint". Suitable extractor/descriptor algorithms are Scale Invariant Feature Transform (SIFT) and Speeded Up Robust Features (SURF). The keypoints are stored in a data tree and are then provided for the online tracking.

During online phase, features are extracted from the current video frame using the same extraction algorithm as in the training phase. These features are then compared to those stored in the provided feature tree. For verifying a match, also nearby features have to match accordingly. If a threshold amount of matches is found in a video frame, its corresponding camera pose can be estimated (Lima et al. 2010:2, 9 sq.).

Concluding, this technique could track strongly textured, flat surfaced props of the MRE. But for guaranteeing constant tracking, there must always be at least one tracking target present in the camera's field of view. Walls of exhibition halls usually feature large explanatory wall texts and graphic material, paintings, etc. It cannot be guaranteed, though, that such targets of a certain size and quality are spread sufficiently over the room.

Concluding, also keypoint based tracking could hardly be used as a standalone tracking method.

I) Conclusion

Above analysis has shown that neither the edge nor the texture based tracking techniques could be used as stand-alone tracking solutions for the MRS: Edge-based tracking has to be initialised on certain object viewpoints and texture based tracking might not have enough targets available for ensuring permanent tracking.

Pressigout & Marchand (2007) and Vacchetti et al. (2004a) propose to combine edge and feature based tracking methods. Such a hybrid approach could exploit both feature types for tracking and hence show higher robustness.

This approach would also be suitable for the MRS: If the system is able to track both textured and contoured objects, more objects qualify as trackables. Therefore, the probability is higher

that there is always a trackable target present in the field of view. If both systems have trackables available at the same time, the overall tracking should be very robust.

Edge based tracking would still require initialisation. But as the texture based tracking component does not, it can be assumed that once a first texture target is tracked and a corresponding camera pose is reproduced, this tracking information could be used to initialise also the edge-based component.

3.2.1.2 Available tracking implementation possibilities

Metaio and Qualcomm Vuforia provide SDKs for creating real time mixed reality applications. They are based on computer vision libraries that implement above methods. Both SDKs are available as extensions to Unity.

I) Metaio SDK

The Metaio SDK offers a large variety of tracking techniques: Among others, markerless 2D image texture based, CAD-model (PS edge based), 3D map (SfM) and instant (SLAM) tracking methods are available (Metaio Developer Portal 2013).

I.I) Edge-texture based hybrid method

As discussed before, more robust tracking results could be achieved by combining texture and edge based tracking – and the Metaio SDK offers both these methods.

It turns out, however, that the SDK's structure does not allow the use of different tracking methods at the same time (Sommer 2013):

The XML tracking configuration, in which the desired tracking method is set allows only one tracking method to be specified. Moreover, only one tracking configuration can be implemented during runtime. Therefore, the Metaio SDK does not offer the desired hybrid tracking solution.

I.II) Edge and/or texture based methods

The Metaio's tracking methods would thus have to be used individually. For the SDK extension to Unity, the following workflow applies, for both texture and edge based tracking alike:

As training data, the SDK's texture based method requires 2D images of flat target objects. The edge based method requires a wireframe model of the target.

In Unity, a project has been prepared with the Metaio extension and the target files have been disposed in the project's "streaming assets" folder.

From the Metaio assets, an "MetaioSDK" prefab object instance is added to the scene.

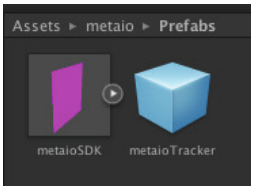

Fig. 36: Metaio SKD prefabs in Unity

This object serves as an overall tracking manager together with the XML tracking configuration: The tracking manager object contains "tracker" child-objects. They all have to be specified in the XML file with one target image, an own coordinate system and their position in it. Consequently, the arrangement of the trackers in the scene is done in the XML configuration. The trackers do not have a mesh representation in the scene. Consequently, their final arrangement is not visualised in the Unity, making the scenario design more difficult than with the Vuforia solution.

Virtual objects for augmentation can be added to the scene but have to be assigned to a parenting tracker. Each object can only be assigned to one single tracker.

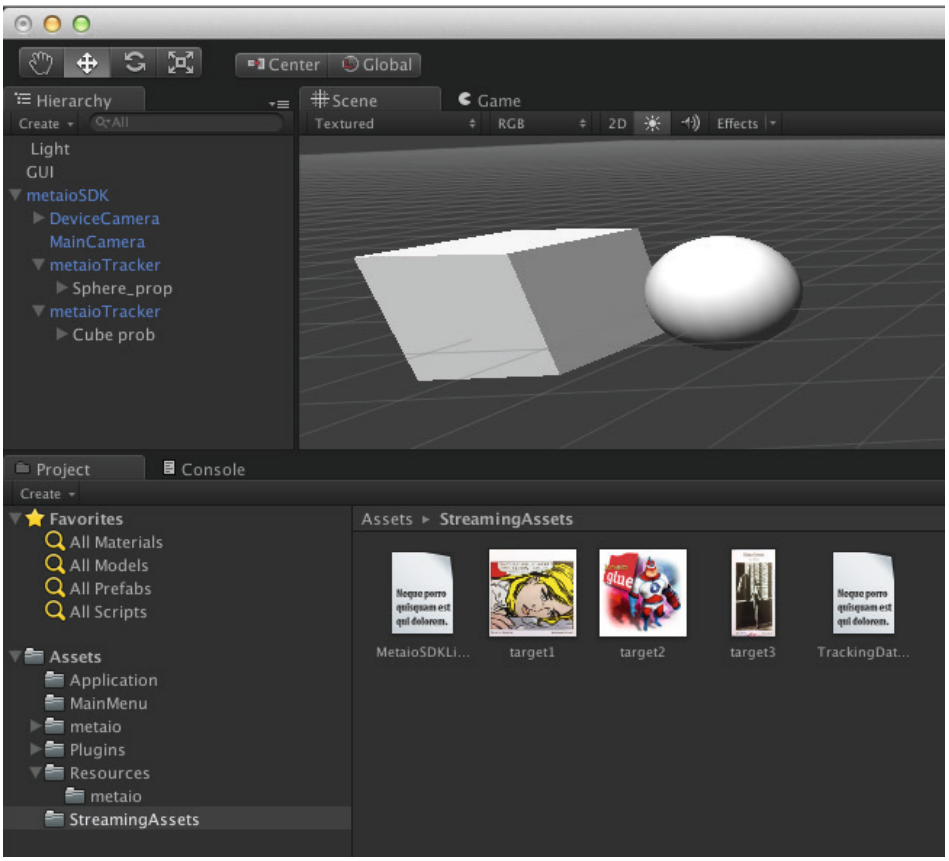

Fig. 37: Prepared Metaio scene set-up and related assets

For the online phase, this implies that if a target is tracked, not the whole scene is rendered but only its assigned virtual probs. Or: A certain set of virtual props are only rendered if their parenting tracker target is (well) visible in the camera's field of view.

For the MRE, this single-target dependency is infeasible for various reasons. For naming two examples: As the virtual props are supposed to be placed IN the MRE' space, they should be visible from every user standing point. But if a centrally placed prop can only be seen as long as one specific target is visible in the field of view, it can consequently only be seen from one side.

Moreover, virtual characters and moveable props can change their position and would therefore simply vanish once they move out of the viewing range of "their" target.

Duplicating all virtual objects for assigning them to all trackers is not an option: Every duplicate would mean a new object instance. On runtime, various instances of an interactive object would behave differently if their behaviour were based on a random variable. As an exemplary outcome, a staged virtual character might suddenly split into multiple.

Concluding, the tracking implementations of the Metaio SDK are impracticable for the purposes of the MRS.

II) Qualcomm Vuforia SDK

The Qualcomm Vuforia SDK offers texture based tracking using 2D image resources. While Qualcomm does not disclose Vuforia's precise methods and algorithms, it shows strong similarities to the analysed keypoint based tracking technique as can be seen from the following workflow:

II.I) Offline phase

Photos of flat target objects and their dimensions are uploaded to a "Target Manager" server, where they are analysed for unique natural features. Dependent on the number of extracted features and their coverage of the target image, the target quality is indicated via a five-star rating scheme. Processed tracking data of multiple targets can be downloaded as a Unity asset package, consisting of a DAT file containing the features' position on the target images, JPEG texture file and an XML listing all the package's target names together with their dimensions.

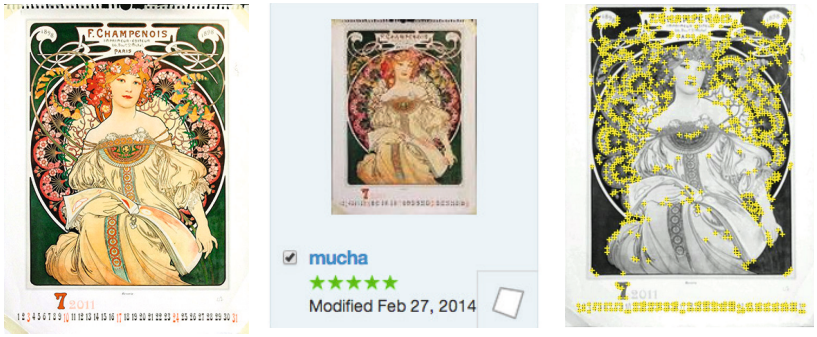

Fig. 38: Target image, quality assessment and found features

In Unity, a project has been equipped with the SDK extension and the asset package's data set has been imported. From the related assets, an ARcamera and an ImageTracker prefab (= template) object instance are added to the scene. The ImageTracker contains an empty plane as a placeholder. In its script component, it is assigned one of the image targets from the target data set. The placeholder plane then resizes to the target's dimensions from the XML target list and loads the corresponding JPEG image texture. The readily prepared tracker can now be arranged in the 3D environment without restrictions.

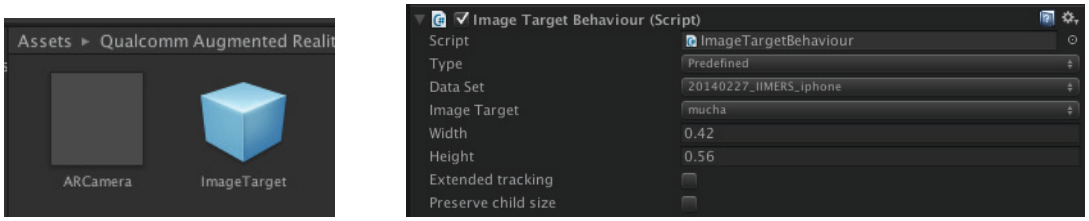

Fig. 39: Vuforia prefab assets and Image Target prefab set-up

Fig. 40: Arrangement of prepared Image Target prefab in Unity scene

After all image targets have been prepared as ImageTracker objects, they can be arranged in the scene according to their relative real-world positions to each other. Consequently, this assigns them their 3D position in the overall object model, completing the required feature descriptor data.

Following, miscellaneous other virtual objects are added to the scene, e.g. simple cubes and spheres. The prepared Unity project is deployed to a mobile device.

II.II) Online phase

When running the Unity app, the phone camera's video stream is output on its display. On pointing the camera at one of the known real-world image target, the target is recognised and tracked. From the perspective distoriton of the target in the video image, the corresponding camera pose to it is calculated. The scene's virtual camera is adjusted to this pose and its output is superimposed upon the real camera's video stream. In the final output, the depiction of the virtual props to the real-world target matches their scene position to the corresponding ImageTracker object. The MRE is thus established.

II.III) Resource quality and resulting accuracy, robustness & performance

Optimal tracking results are first and foremost leveraged by the resources' quality, i.e. the target images and phone camera's video stream.

Combining the stated quality criteria for natural texture features with the image target guidelines by Vuforia (Qualcomm Vuforia Developer Portal 2013a; Qualcomm Vuforia Developer Portal 2013b), the source images should

show	not / little show
• a generally high image quality (sufficient resolution, wide range of contrasts, sharpness) • fine structures • high local contrast • high feature distribution over the whole image	• smooth areas. These do not provide features • organic shapes, e.g. curves as they cannot reliably be tracked • repetitive patterns. The features have to be unique to be assigned correctly

Table 2: Target quality criteria

As regards suitable targets in sites of cultural heritage, the criteria for suitable targets apply for filigree paintings, wall texts and graphics, decorative textures, etc.

The video stream quality is controlled by the quality of the camera (e.g. its resolution, dynamic range) and also by the lighting conditions of the environment. The room should be sufficiently and evenly lit in order to enable optimal tracking conditions.

If optimal conditions are provided, the tracking accuracy and robustness are sufficiently high, due to the fact that the same video frame that is used for the feature extraction is used for superimposing the correspondingly calculated virtual video frame. But as the tracking data is newly calculated with every frame, the resulting camera pose can therefore differ slightly even when not moving the camera. Therefore, the output tends to jitter.

Vuforia is generally geared to provide real time performance on mobile devices, i.e. between 15 and 30 fps. In order to ensure best performance conditions, the SDK's parameters should

be optimised. As regards the SDK's direct parameters, threshold values or other parameters of the tracking quality cannot be manually altered. But the performance can indirectly be aided by keeping the processing demands of the resources low, e.g. by providing a video stream and target data of reasonable solution and not having more target images loaded (= activated) than necessary.

3.2.1.3 Conclusion

Markerless model based natural feature tracking techniques would generally state a suitable tracking solution for the MRS: They are implemented in-phone and hence do not rely on intrusive external set-ups. The texture /keypoint based technique of the Vuforia SDK delivers usable results: Provided that optimally prepared tracking resources are available, it can provide sufficient registration accuracy at real time performance.

Some of the MRS's frame conditions, though, can only be met to a limited extent:

For allowing highest mobility and a robust MRE illusion, the tracking system must permanently be able to provide data over the space of the whole room. With feature tracking, this cannot be guaranteed. Even if edge and texture based hybrid approaches facilitate to use a wide range of real-world objects as trackables, still "dead spots" would appear: For one, it cannot be guaranteed that any room provides evenly spaced target-material for a basic coverage. Then, generating tracking data from all possible points of view in a room would require that all present textures in the room to the finest detail level would have to be trackable.

Considering the fact that only texture based tracking (Vuforia SDK) is available for the MRS's implementation, the variety of possible targets is already considerably reduced.

Concluding, the presented visual tracking method is a conditionally suitable solution. When using it, it would require a supporting non-markerless tracking system in order to bridge occurring tracking gaps.

3.2.2 Active tracking techniques

Active tracking techniques encompass those for which "external equipment is used to monitor the users movements" by exploiting "sound, light or electromagnetic fields to track where some device on the user is relative to fixed devices. This means that these fixed devices must be rigged up to surround the site, and the will likely need power as well" (Strand 2008:16, 25).

These tracking techniques hence require at least some external hardware set-ups and are therefore intrusive.

But as it has been shown in the previous section, passive in-phone implemented systems do not provide a universal solution for all site types. Therefore, also active systems have to be taken into consideration, either as a secondary system to a passive one or as a stand-alone solution. Their intrusiveness would thus state a trade-off to providing a universal and comprehensive tracking solution.

3.2.2.1 GPS and dGPS

The iPhone features a built-in Assisted GPS receiver (Apple 2013b) which uses a satellite-based navigation system for retrieving its location. As this might therefore prove as a "ready to use" in-phone tracking system, this approach should be analysed for its suitability for the MRS:

The GPS receiver uses the satellites' timestamped radio signals to calculate its distance from them via the signals' time of flight. The phone's position can subsequently be calculated via

trilateration (Kyle 2012; TomTom 2013). GPS delivers data accuracy of 15 metres (Garmin n.d.), which is clearly not sufficient for the MRS's purposes.

Differential GPS (DGPS) uses extra reference stations, the precise location of which is known. Their correction signals are then used to calculate more accurate localisation data (Seeber and Schmitz n.d.). The baden-wuerttembergian Land Surveying & Geo-Information Office offers such a positioning service (SAPOS). Their HEPS (high precision real time positioning service) delivers real-time positioning data of 2-3 cm accuracy (SAPOS n.d.). This accuracy could prove to be sufficient for the MRS.

However, the GPS and DGPS signals alike are not strong enough to reliably penetrate walls and objects, making them unsuitable for indoor use. For this reason, DGPS does not qualify for use in the MRS.

3.2.2.2 Wi-Fi-based localisation system

Wi-Fi-based positioning systems leverage the signal strengths of Wi-Fi access points for real time localisation (Bahl and Padmanabhan 2000). This could basically be implemented in the MRS because the iPhone has a built-in Wi-Fi antenna and many public buildings provide a network of several Wi-Fi access points. Subsequently, this tracking system could use already available external infrastructures and would thus not be further intrusive. These characteristics make this system a particularly attractive solution approach.

Using this system requires a prior training phase, in which signal strengths at certain positions of the room, so-called "fingerprints" are collected. In the online phase, the system produces a current signal strengths sample and searches its equivalent in the "fingerprint" map, thereby determining the device's position (Liu et al. 2012:2).

Assumably, a fingerprint map of many dense samples leads to more accurate localisation results. Tests by Liu et al. (2012:3 sq.), however, do not prove this: For one, the signal strengths of the access points do not differ significantly over short distances. This basically limits the density of the fingerprint map to sample of those distances where strengths differences are significant enough.

Moreover, the signals can be distorted by multipathing occurring particularly near walls as well as by the hand of the user covering the device's antenna. Additionally, samples of different locations can also happen to show the same signal strengths. These factors lead to wrongly assessed fingerprints in ca. 10% of the cases.

As a result, the achieved average accuracy ranges between 3-4 metres with a 10% error rate of 6-8 metres difference to the actual position (Liu et al. 2012:1 sqq.).

Concluding, this system is convenient to use as it utilises broadly available infrastructures while its accuracy and error rate would only be sufficient for room-level tracking and consequently not for the MRS.

3.2.2.3 Peer assisted acoustic ranging

This localisation system uses acoustic signals exchanged between nearby phones that thereby estimate their distance to each other (Liu et al. 2012:3). "First, the two devices will each in turn emit a specially designed sound signal, called a "Beep", within one second. Meanwhile, each device will also record a few seconds of continuous sound from its microphone. Each recording should then contain exactly two Beep signals picked up by its microphone, one emitted from the other device and one from itself. Next, each device will count the number of sound samples between these two Beeps, and divide the number by the sampling rate to get the elapsed time

between the two TOA events. The devices further exchange the elapsed time information with each other. The differential of these two elapsed times represents the sum of the time of flight of the two Beeps and hence the two-way distance between the two devices." (Peng et al. 2007:2).

The "Beep Beep" system presented by Peng et al. (2007; 2012) implements this with basic commodity cell phones, achieving a ranging accuracy of 0.8 cm. This approach is interesting for the MRS as its external hardware would be small enough to not be visibly intrusive in a site and surely because of its high accuracy.

It is not clearly stated, though, if this system can perform at real time speed.

Moreover, the MRS' frame conditions require that the frequencies of the "beeps" would be in the non-hearable range in order not to be intrusive. Cell phones' microphones and speakers, however, are optimised for the frequency range of the human voice. This implies that "beeps" outside this range would most probably neither be emitted nor recorded at a reliable quality, resulting in less general accuracy and a higher error rate. The "beeps" therefore will have to be of a frequency within the human voice range – and hence be hearable.

As such a constantly hearable sound would not only be intrusive for the site but also annoying for its visitors, this localisation system cannot be used in the MRS.

3.2.2.4 RFID systems

These RTLS use radio frequency (RF) emitter-receiver combinations.

The two technologies presented here, Ubisense and Active Bat, employ portable tags emitting an RF signal which is received by fix installed sensors (also referred to as beacons) (Cambridge University Computer Laboratory 2005).

The Ubisense system uses UWB pulses that are emitted by its portable UbiTags. These pulses are received by fix installed sensors (at least two) (Ward 2005:1).

Based on the time-of-arrival differentials of these signals, trilateration is used to assess the position of the individual tags. Additionally, also the signals' angles of arrival are measured and evaluated by triangulation methods, making the results more accurate (Aitenbichler et al. 2009).

As regards the required sensor density, the mixed reality installation "Traffic" (Lintermann et al. 2011), was realised using the Ubisense system and managed to cover a large area of ca. 750 square metres with not more than six sensors. Due to the efficient coverage capabilities of this system, its intrusiveness could still be acceptable as regards the stated site requirements.

The achievable accuracy of the system, however, is 17 cm at an error range of 10% as testing data by Ubisense shows (Ward 2005:9).

Concluding, the system would not be very intrusive as it would not require many sensors to be installed on the site but the provided accuracy is far not sufficient for the requirements of this project.

The ActiveBat system presented by the Cambridge University Computer Laboratory works in a similar way as the Ubisense system (active tags and fix installed receivers; trilateration) though achieves an accuracy of ca. 3 cm. This could be sufficient for the purposes of the MRS. This system, though, requires a sensor density of 1.2 metres distance between the single beacons. As an example, this corresponds to a grid of 720 receivers on 1000 m^2 (Cambridge University Computer Laboratory 2005). Although this system features suitable localisation accuracy, it would be way to intrusive for being installed on a site of cultural heritage.

As a summary, both RFID systems would not be suitable for being used in the MRS as they are either not accurate enough or intrusive.

3.2.2.5 Infrared Motion Capture systems

Infrared motion capture (IrMoCap) systems use a set of IR-cameras with IR-emitting diodes placed around the camera lens. Their light is reflected by retro-reflective markers in the room and captured by each camera in return (Skogstad, Nymoen, and Høvin 2011:2). "The retro-reflective markers appear significantly brighter than any other object in the camera view, and image processing (thresholding and circle fitting) is used to track those markers in each view. Triangulation of multiple camera views results in very accurate and robust 3D marker tracking" (Bregler et al. 2005:3). Test results by Skogstad et al. (2011:3) indeed show drift-free tracking accuracy of millimetre precision at real time performance. These characteristics would meet the core requirements of the MRS.

That this system can be used for moving in a virtual environment by the help of a tracked device has been shown by VFX studio WETA digital:

In a large studio, a visual motion capture system was used to track a video camera equipped with a rig of reflective markers. The tracking data was fed to a scene camera in a virtual environment. The virtual camera's viewport is then output back on the "real" video camera's display in real time, creating the illusion of being able to move & film within the virtual environment (The Hobbit: Production Diary, vol.13 2013, 07:19 – 08:15). For the implementation of the MRS, a similar set-up could be targeted.

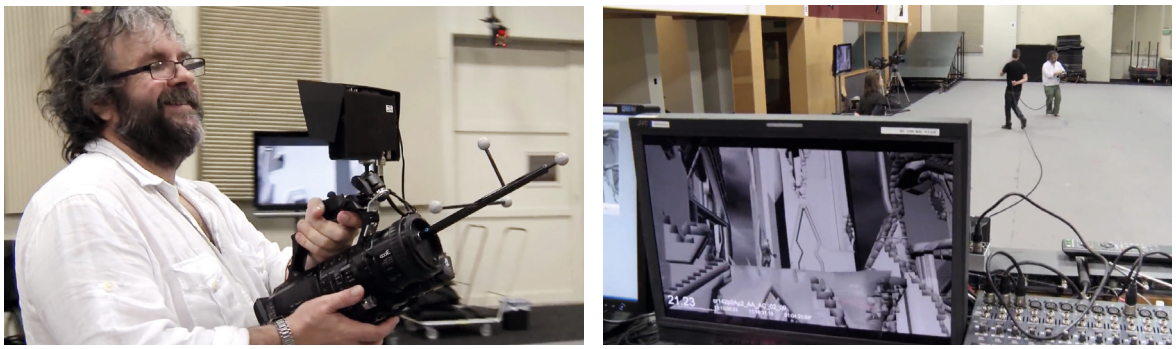

Fig. 41: WETA VR Motion Capturing set-up (The Hobbit: Production Diary, vol.13 2013)

Consequently, this technology is highly interesting solution approach for the MRS. Its precise suitability will therefore be analysed in detail.

I) Tracking quality

The quality of the tracking data is controlled by several parameters:

I.I) System calibration

The first crucial factor is the quality of the system's calibration, a process during which the "position of the cameras in relationship to each other, and in relationship to a global coordinate system defined by the user" (Skogstad, Nymoen, and Høvin 2011:2) are determined. Poor calibration can lead to distorted tracking results and should therefore be avoided (VICON 2002:16).

I.II) Camera resolution

Another factor is the resolution of the camera sensors: The higher their capability of optical differentiation, the more accurate the tracking results (Skogstad, Nymoen, and Høvin 2011:2). This is strongly correlated to how large / well-defined the reflective markers appear in the camera view: The further away a marker from the camera or the wider the camera's field of view, the more difficult for the system to assess its precise position.

It would therefore be important to make sure that the camera sensors provide enough resolution over their FOV to reliably track a marker also at the opposite end of the room.

Fig. 42: Examples for available IrMoCap resolutions and frame rates: VICON T-Series (VICON, 2013)

In case this cannot be achieved by the cameras, Bregler et al. (2005) suggest to use larger markers (than standard 0.5 inch) in large tracking volumes. This trade-off would compromise on the overall accuracy of the system – but therefore reduce tracking errors considerably.

I.III) Tracking volume & marker rig

Constant 3D tracking can only be provided if the markers are always within view of at least two cameras (VICON 2002:16). And as stated in the frame conditions, the tracking volume should cover the whole room in order to allow the user to move around as freely as possible. If markers are occluded or outside the tracking volume, the tracking system cannot provide current data for the proper MRE rendering, consequently breaking the MR illusion.

It is therefore important to equip the room with enough cameras to provide a well covered tracking volume.

Standard IrMoCap setups aim at covering a central area of a room (i.e. ca. 50% its total size) by placing eight equally spaced cameras as visualised below:

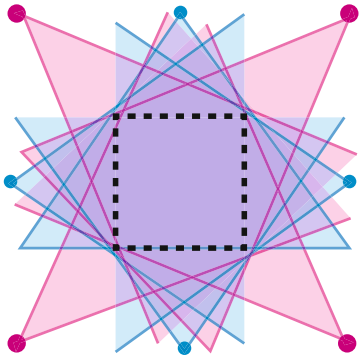

Fig. 43: Standard set-up as suggested by VICON: Four 45° FOV (red) and four 70° FOV (blue) equally spaced cameras (VICON 2008:14)

For the purposes of the MRS, the coverage could be optimised by using additional angled cameras as visualised below:

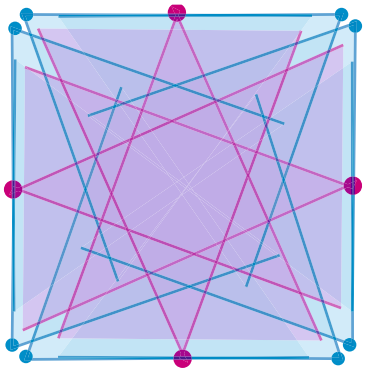

Fig. 44: Set-up for full room coverage: Four 45° FOV (red) and eight 70° FOV (blue) equally spaced cameras

II) Retrieving an objects position and orientation

As shown in the example of WETA digital, markers can be assembled on a rig, mounted on top of the mobile device. By using several markers of a non-symmetrical formation, the tracking system could also provide the vital information of orientation and rotation of the mobile device. This would make the use of additional sensors, e.g. inertials, obsolete.

III) Data integration in Unity and latency

For calculating and integrating the tracking data of the IrMoCap system into Unity, the following data processing workflow applies: The raw images of the IR-cameras are sent via gigabit Ethernet to a central processing software that calculates the corresponding 3D tracking data. The prepared data is then made available for further integration via data streaming. On the receiving device, the data stream is read and fed into Unity's scene camera.

As stated in the development frame conditions, the end-to-end latency of this process must be as low as possible and never above 33 ms (i.e. 30 Hz minimum sampling rate) in order to ensure real time performance. Moreover, the overall system should stay cable free, requiring wireless data transfer.

This can be achieved using the following solution: VICON provides the software Tracker, which is capable of processing the cameras' data ready for transfer at a latency of minimum 2.5 ms. For the streaming of the data, Tracker allows the use of the UDP and TCP protocols and the built-in VRPN server-side interface. The latter is specially geared for integrating streaming data of external peripherals into VR applications (Taylor II et al. 2001). The data can be transmitted wirelessly using the IEEE standard 802.11n at 2.4 GHz (also known as Wi-Fi), which is supported by the iPhone 5 (Apple 2013b). Once the phone has received the streaming data, these can be directly integrated into Unity using VRPN C#-Wrapper Unity extension provided by the UART project group of the Georgia Institute of Technology (MacIntyre et al. n.d.).

As the data transfer volume is supposedly far below 1 KB per sample, the suggested transfer and integration workflow should be fast enough to not add up significantly to the 2.5 ms processing latency of Tracker.

IV) Conclusion

Using an IrMoCap system for tracking would provide the required tracking data quality at real time availability, leveraging accurate registration and smooth movement reproduction.

The cameras could be set up to cover the whole room and the tracking data could be wirelessly integrated, allowing constant reliable tracking while the user can explore the MRE.

So far, this approach meets the core requirements of the MRS best.

However, some side requirements are not met: The system requires the set-up of twelve external cameras in the room. Even if these are usually located on the ceiling, they might still appear intrusive and their red light distractive for visitors. Furthermore, an additional workstation would be required for processing and streaming of tracking data. This might be considered intrusive as well as it cannot be guaranteed that it can be hidden from the visitors' sight e.g. in a neighbouring room. The intrusiveness of the additional marker rig on the mobile device can be neglected. Moreover, it has been stated that some sites could have restrictions on the use of infrared light due to its damaging effect on sensitive artefacts. In some sites, it might therefore not be possible to implement the MRS with an IrMoCap system.

This solution is therefore suitable but not universal as it might only be used on some sites with corresponding set-up and lighting policies.

3.2.3 Conclusion

The analysis has shown that from all the evaluated tracking technologies, the infrared motion capture system is technically the most suitable solution for achieving the required result quality: This system is capable of providing permanent tracking data of millimetre accuracy at full room coverage.

However, the system has also been shown not to be a universal solution as for its use of intrusive external set-ups and infrared light, it might not be applicable for some sites.

The secondary solution is the use of a natural-feature visual tracking system: This system delivers reliable tracking results while not suffering from the IrMoCap system's shortcomings for being implemented all in-phone. On the other hand, it has also been shown that neither this tracking system can be used as a universal solution: It cannot be guaranteed that any site provides enough evenly spread targets so that there would always be a trackable in the camera's current field of view. For covering the "dead" tracking spots, further externally added markers would have to be added – which again would state an intrusive measure.

Concluding, the implementation success of the system depends entirely on the conditions of the site: The "best case" solution of an entirely in-phone implemented system can only be met if the site in question provides a sufficient density of useable targets so that only natural feature based tracking could be used. If this condition is not given, "middle case" solutions apply, meaning that the system relies on additional external set-ups: If the site allows the use of infrared light, an IrMoCap system is the solution of choice. Otherwise, the site's provided textures can be used for natural feature tracking and additional markers would have to be added in order to ensure permanent tracking. The additional markers could be designed to fit well into the given environment in order to diminish their intrusiveness.

Should the site neither provide a sufficient target density nor allow the installation of intrusive set-ups, the system cannot be implemented – which would state the "worst case".

4 PROTOTYPE IMPLEMENTATION

The following section documents the implementation process and performance testings of a prototype system, which will be realised with those implementation possibilities found suitable in the previous analysis. The prototype testing aims at proving if an effective, usable output result can be achieved with the found methods as well as to establish implementation difficulties.

For providing representative testing conditions, the prototype will be realised in a testing environment that shows typical features of a site of cultural heritage. The 3D retrieval of the testing site as well as the tracking implementation will be conducted with available methods and technologies. The MRE will be equipped with rich virtual content and interactive behaviour in order to provide the computational load of a typical scenario. The readily prepared MRS will then undergo a test run, which will provide concrete evidence on the system's computational performance and the quality of the tracking result.

4.1 AVAILABLE METHODS AND TESTING ENVIRONMENT

For the implementation of the prototype system, the following technologies and methods are available:

The MRE will be realised in Unity Pro 4.3.2f1. An iPhone 5 (2013) with iOS 7 and standard technical configuration is available as a testing device.

For the retrieval of the general architecture, neither blueprints nor large volume 3D scanning technologies are available and will therefore be by manual measurements, i.e. using a laser distance meter and yardstick. Smaller objects of furniture can either be digitalised using the KinectFusion 3D scanning method or manual measurement.

Regarding tracking technologies, only visual tracking is available for the prototype testing. This will be realised using the Vuforia SDK 2.8.7 extension for Unity and a Sony Alpha 55 DSLR camera with a $f/2.8$ 24-70 mm zoom lens, which serves for retrieving the target images.

The testing site is a warehouse hall of 12.33 x 11.97 x 4.5 metres, originally build in 1935. Today, the room is used as a media production studio and architecture model storage. The room's original architecture features rough, undecorated walls, a slightly uneven wooden floor, a window side, two carved-out doorways and four roof-supporting pillars in its centre. For its today's use, it has been added a wall-sized metal storage-shelf and several worktables as well as chairs and lamps.

The exemplary site is suitable as a testing environment as it shows the typical features of exhibition rooms in sites of cultural heritage: While the old-style architecture of the room can be understood as an original room of a cultural site, the shelves and tables with their architecture models resemble typical exhibition furniture displaying artworks and artefacts.

For further preparing the room in the style of a common exhibition venue, posters of photos, graphics and texts are added to represent exhibited artworks and explanatory wall-texts. Furthermore, a mannequin, representing a statue, is positioned in the room in order to provide an exemplary non-flat-surface occlusion object.

Fig. 45: Test site preparation

4.2 IMPLEMENTATION PROTOCOL

4.2.1 Phase I: Setting up the MRS

4.2.1.1 Reproducing the real-world environment in Unity

I) General architecture

All permanently installed elements of the room that could state occlusion objects or obstacles for the virtual population were retrieved by manual means and reconstructed in Unity. Included elements encompass the walls and doorways, the pillars, the pipeline rag at the window side and the metal shelf. The latter two were reconstructed using only primitive block shapes as they mainly provide obstacles for the virtual population but do not require a higher detail level for the storytelling. The reconstruction excludes the right-hand window cutouts as well as the ceiling are these do not provide obstacles or occlusion objects.

On measuring the site, it becomes clear that the room's construction is very uneven: The walls are not positioned to each other in perfect right angles and also the floor level differs by ca. 3 cm over the whole room area. Also the pillars are not fully orthogonal to the floor and show uneven shaping.

These irregularities can barely be reconstructed in the virtual replica by using means of manual measurement. It was therefore not possible to reconstruct the room as accurately as targeted.

It will be shown in the test run how much these discrepancies between the real and the virtual room affect the registration quality.

II) Furniture and other objects

The tables, for being flat-surfaced objects, could also be digitalised using manual means. The mannequin, as a non-flat-surfaced object, was digitalised with the help KinectFusion 3D scanning. The mannequin is the same object that was used in the Kinect feasibility analysis, which has shown that this object could be retrieved at a sufficient quality.

III) Lighting situation

For reproducing the lighting situation, a directional light was set to represent the sunlight shining from the window side. The window side was roughly reconstructed using primitive shapes, in order to cast corresponding shadows onto the scene. The electrical lamps were reproduced using point lights. Together with the ambient light, all light sources were balanced to resemble a standard daylight situation.

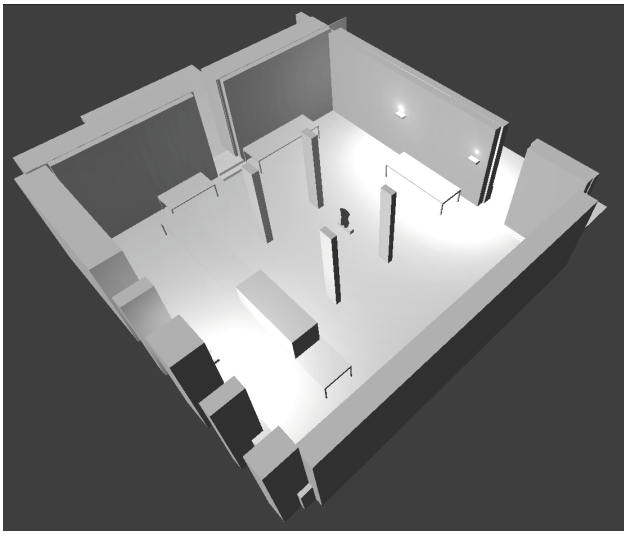

Fig. 46: Reconstructed room architecture and lighting situation

IV) Camera matching

The matching of the virtual to the iPhone 5's camera is handled by Vuforia: On runtime, Vuforia calculates the required camera FOV "based on the intrinsic camera calibration of the specific device (coming from the device profile)" (Qualcomm Vuforia Developer Portal 2013c) and sets the virtual camera parameters accordingly.

4.2.1.2 Setting up the tracking system

All posters were photographed from an orthogonal point of view with a DSLR camera at f 50 mm. The images were undistorted by using the related camera lens' profile in Adobe Lightroom and Adobe Photoshop's and their white balance was corrected. The prepared images were uploaded to the Vuforia Target Manager in order to extract their texture features. The target manager assessed good trackability for all posters. The processed Vuforia tracking data was arranged in Unity according to the poster's position in the real-world environment.

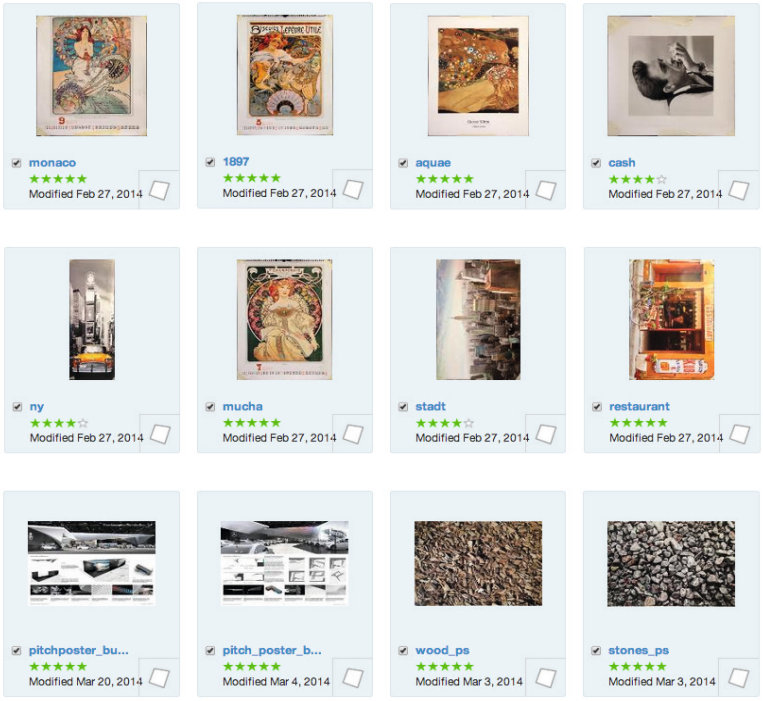

Fig. 47: Vuforia Target Manager view of all uploaded target images

Fig. 48: Arranged targets in Unity scene (excerpt)

4.2.1.3 Designing the scenario contents & menu elements

I) Props

For reproducing the setting of a warehouse, the virtual replica was equipped with a wide range of related props. The props have been set up as rigid bodies and added related sound design that is played on collider triggering.

Fig. 49: Unity scene equipped with virtual props

II) Characters

Four virtual characters were added to the scene and equipped with a multitude of animations and interactive behaviour: For one, the characters have been staged to walk to randomly picked target objects in the room and "work" on them for a certain amount of frames before heading to another randomly picked target object. The characters have further been added a trigger collider, which has been set up as a listener for the following events:

If a character collider is triggered by a collision with another character's collider, a waving animation and "hello" sound is played, representing a greeting action.

If a character collider is triggered by a collision with the camera's collider, the characters will pause in their "work" and instead turn to the camera. The character's audio source component will play a corresponding sound file, representing "talking" to the user. Once the camera exits the character's collider, the character resumes his "work" (for example, see enclosed video B0).

Fig. 50: The virtual population

All the virtual objects' meshes and textures have been adapted to a reasonable resolution and the suggested rendering optimisation measures (see section "Rendering optimisation") have been applied to the scene.

III) Virtual camera

The virtual camera was equipped with kinematic rigidbody behaviour and a NavMeshAgent obstacle component in order to incorporate it into the MRE as a physical object for props and characters alike.

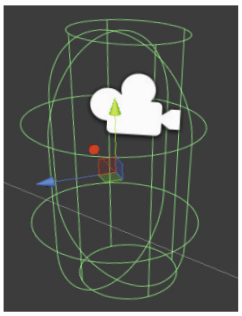

Fig. 51: Main camera with rigidbody collider and NavMeshAgent obstacle collider

IV) GUI

As regards the GUI menu elements, the user is provided a start screen, a help menu and a recovery-instruction screen. The recovery instruction screen is of particular importance for the prototype testing, as it gets activated if Vuforia cannot detect any target in the camera's current field of view. It therefore helps to establish if some targets can only be tracked from a certain distance or cannot be detected at all.

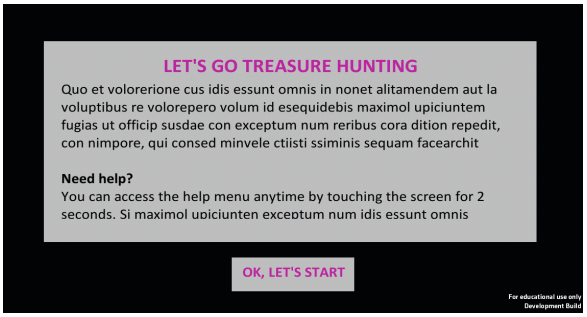

Fig. 52: Start screen

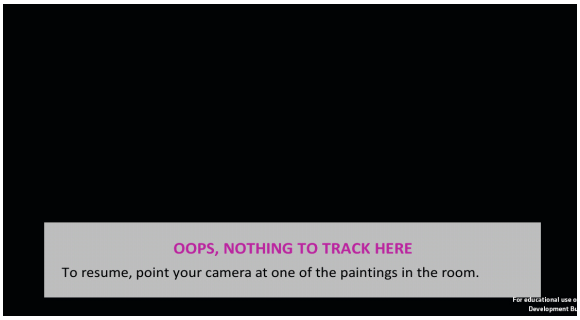

Fig. 53: Tracking failure screen

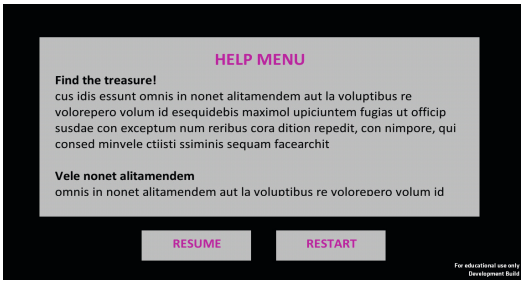

Fig. 54: Help menu screen

Vuforia handles the creation of the composite output. The readily prepares Unity scene is deployed to the iPhone 5 for conducting the test run.

4.2.2 Phase II: Performance testing and tracking quality assessment

4.2.2.1 System performance

The system performance during the test run was monitored via the provided Unity statistics. The statistics show that the overall achieved update rate never fell below 28.5 fps throughout the duration of the test run. Also visually, no slacking or frame freezing was observed. The prototype therefore delivered stable real-time results.

The statistics also reveal that the achieved computing and graphics latency (maxima observed: 12.9 ms on CPU, 1.7 ms for rendering) actually lie far below the final update time of the composite output (average 33.3 ms, equalling 30 fps). The final update rate, though, cannot exceed 30 fps due to the fact that the iPhone camera video maximally supporting 30 fps.

```
---------------------------------------
iPhone Unity internal profiler stats:
cpu-player>    min:  4.5   max:  7.5   avg:  6.0
cpu-ogles-drv> min:  0.1   max:  0.4   avg:  0.2
cpu-present>   min:  0.4   max:  2.6   avg:  0.8
frametime>     min: 29.6   max: 35.6   avg: 33.3
draw-call #>   min:   5    max:   5    avg:   5     | batched:
0
tris #>        min:   178  max:   178  avg:   178   | batched:
0
verts #>       min:   404  max:   404  avg:   404   | batched:
0
player-detail> physx:  0.5 animation:  1.4 culling  0.0
skinning:  0.0 batching:  0.0 render:  2.0 fixed-update-count:
1 .. 2
mono-scripts>  update:  1.6   fixedUpdate:  0.0 coroutines:  0.0
mono-memory>   used heap: 589824 allocated heap: 786432  max
number of collections: 0 collection total duration:  0.0
---------------------------------------
```

Fig. 55: Test run profiler statistics 1: Minimum values recorded when no target was tracked and the MRE was hence not rendered

```
---------------------------------------
iPhone Unity internal profiler stats:
cpu-player>    min:  9.2   max: 12.9   avg: 11.0
cpu-ogles-drv> min:  0.6   max:  1.7   avg:  0.8
cpu-present>   min:  0.6   max:  2.7   avg:  1.0
frametime>     min: 29.7   max: 35.3   avg: 33.2
draw-call #>   min:  39    max:  69    avg:  57     | batched:    20
tris #>        min: 65488  max: 77997  avg: 70259   | batched:   250
verts #>       min: 36683  max: 62395  avg: 46389   | batched:   500
player-detail> physx:  0.5 animation:  1.3 culling  0.0 skinning:  0.2 batching:  0.2 render:  3.6 fixed-update-count: 1 .. 2
mono-scripts>  update:  1.6   fixedUpdate:  0.0 coroutines:  0.0
mono-memory>   used heap: 614400 allocated heap: 786432  max number of collections: 0 collection total duration:  0.0
```

Fig. 56: Test run profiler statistics 2: Maximum values recorded during interaction with virtual props and maximum of vertices in field of view

4.2.2.2 Tracking & registration quality

I) Target detection and tracking volume quality

On testing the trackability of the targets, it was established that the images were not tracked as stably as would have been expected from the high-quality training data, even if the camera was positioned close to a target image.

The training data preparation was repeated by photographing the target images anew using the iPhone 5's camera. On testing these tracking data, Vuforia delivered fast detection and far more stable tracking results. It can be assumed that this is due to the high resolution DSLR images (16 MP), which provide mostly high-detail features that cannot be detected in the video stream of the iPhone 5's camera, which provides only half the detail resolution (7.3 MP). The feature detection therefore works more stably if the image training data and detection source provide the same detail-resolution features.

Even though the target detection and tracking has been improved, the first test outputs still show noticeable jitter. The jitter already appears if the camera is positioned close to a trackable and increases the further the camera moves away from it. The jitter indicates that Vuforia is not able to precisely locate the target features in the video source and therefore calculates the virtual camera's position based on erroneous tracking data which differ with every tracking update. This is caused by quality insufficiencies of the camera's video image. As the room is well lit and the camera image shows the target with maximum sharpness and dynamic range, insufficient lighting and camera settings can be excluded as error sources. The reason therefore lies in the limited detail-resolution and the occurring mid-tone noise of the camera image. As the jitter it consequently caused by the shortcomings of fix hardware characteristics, it can neither be eliminated nor diminished.

Although the testing site had been prepared with an above-average density of trackables, permanent tracking could not be facilitated:

It was established that all of the targets, even large ones with prominent features, could only be detected from much shorter minimum distances than expected. It was assumed that the tracking system could track targets also over longer distances as long as their prominent features were well visible in the field of view. The measured minimum distances, though, ranged between 80 cm for the small, 1.5 to 2 metres for the mid-sized and ca. 3.5 metres for the three large targets provided they were detected from an orthogonal point of view. For angled views, even shorter distances applied. The applied targets, however, were all spaced at larger distances. This led to the result that the system could sometimes not produce any tracking data although a multitude of target images was present in the field of view. Adding more targets in order to achieve a full tracking volume was not an option as the resulting target density would not have been realistical for any site.

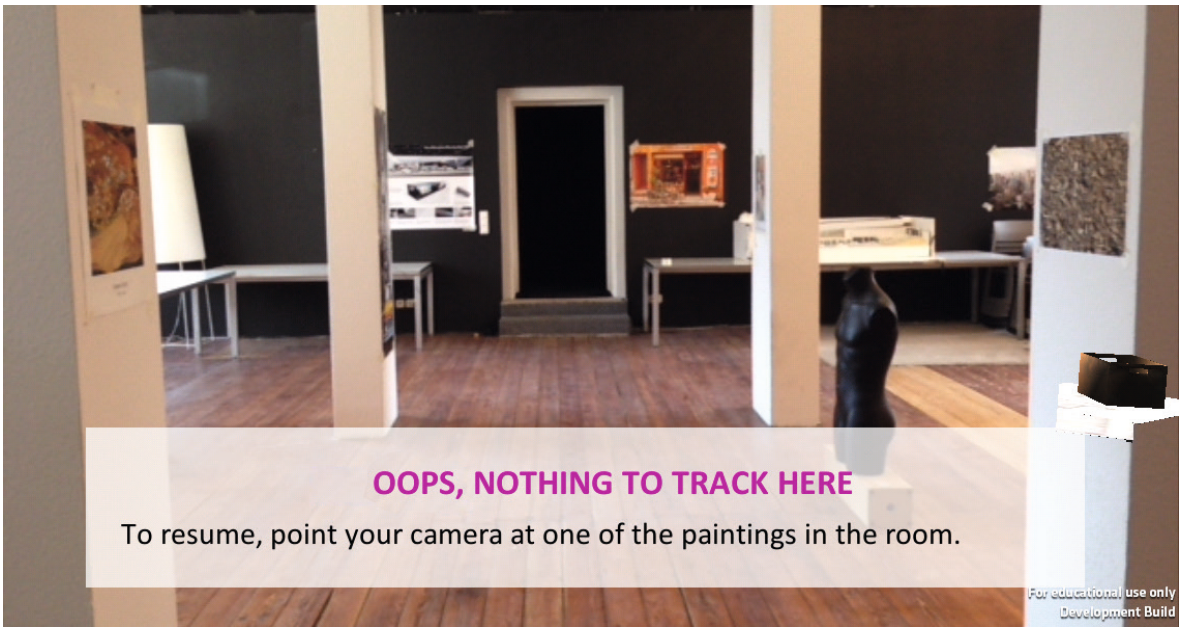

Fig. 57: Although several targets are visible in the camera view, tracking could not be initialised

The reason for the limited detection range of the tracking system is again related to the camera video's resolution and sharpness. The single frames do not provide a sufficient level of detail for detecting the targets over the required distances, which becomes evident in above testing screenshot.

Once the targets where detected, however, they would remain tracked over much longer distances, mostly at least double the target's minimum detection distance.

Another reason for the observed non-permanent tracking is the sensitivity of the camera video to motion blur, which occurs if the user moves the iPhone device fast. Consequently, Vuforia is not provided a sufficient tracking source for extracting features. As a result, the tracking breaks.

Moreover, Vuforia sometimes showed severe tracking errors that did not recover, also not if another target was tracked. The only solution left was to manually restart the app or reloading the level, respectively.

II) Registration quality

II.I) Constant offsets

From viewing positions close to targets, the composite output showed smaller constant-sized registration offsets that could directly be related to geometric discrepancies between the real-world object and their virtual replica model. These offsets can be considered a consequence of the digitalisation difficulties mentioned earlier.

Fig. 58: Examples of constant offsets on two of the pillars

II.II) Dynamic offsets

The MRE shows offsets that increase with the distance from the camera and further increase if the camera is moved. These offsets also show jitter and do not reproduce in the same way when a target is detected anew (for examples, see enclosed videos B6 – B9).

For better visualisation of this phenomenon, the replica was applied semi-transparent materials of different colours, so that the composite output showed the overlapping of the real environment and its virtual replica.

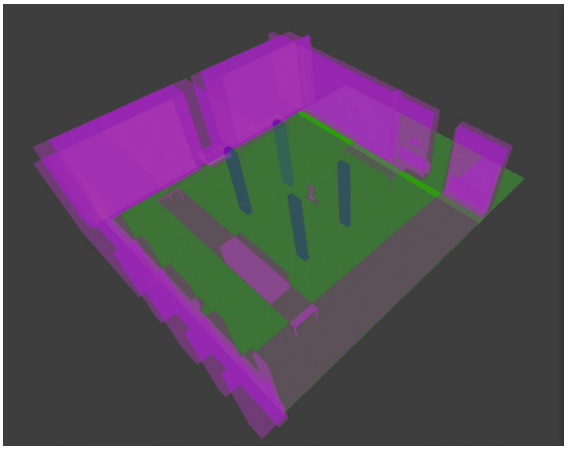

Fig. 59: Room reconstruction with coloured materials

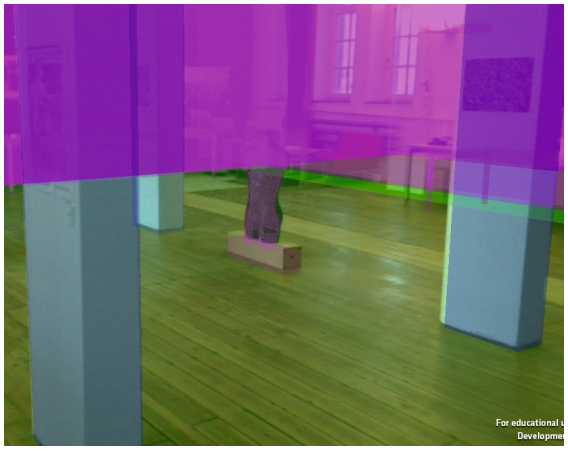

Fig. 60: Composite output

With the help of this testing visualisation it was established that

a) when the camera is hold still, the overlapping shows smaller registration offsets and jitter that increase with distance (for examples, see enclosed video B10). As mentioned before, jitter indicates erroneous tracking data so that the position and orientation of the virtual camera cannot be reproduced accurately. As a result, the cameras show slightly differing viewing angles, leading to the observed offset incrementation over distance. It is not possible to view the output without slight jitter but it can be assumed that the smaller registration offsets also derive from small imprecisions in the virtual model as it is known that a perfect reconstruction was not achieved. It can be excluded, though, that the dynamic offsets are a result of strong discrepancies between the virtual and the real model. It can also be excluded that the offsets derive from poor camera matching as the offsets are only translated but not distorted. The latter would occur if e.g. the fields of view or the lens parameters did not match.

b) when the camera is panned, the registration offsets increase. Assumably, this results from inaccurate tracking data caused by the motion blur of the panning. On basis of the blurred video image, Vuforia cannot reproduce camera movement and rotation accurately. This is confirmed by the fact that the registration offsets decrease again once the camera panning stops (for examples, see enclosed videos B11 – B12).

c) the extent of above effects depends on the tracked target. The small, graphic target images produced less offsets than the large photography prints. This is due to the fact that the features of the photography targets are less well defined than those of the graphical posters and hence more difficult to track. Furthermore, the large posters could not be placed fully planarly to the walls but always showed at least some bulges.

It can therefore be concluded that the dynamic offsets are mainly caused by the insufficient resolution of the iPhone camera's video and by motion blur. Other factors that add to these offsets are imprecisions of the virtual reconstruction model as well as imperfectly installed and less trackable targets.

4.3 EVALUATION

The prototype implementation could not achieve a result that meets the MRS's frame conditions mainly due to lack of permanent tracking as well as to insufficient registration quality.

Permanent tracking could mainly not be achieved because the iPhone camera video's limited resolution did not allow to detect and track targets over the required distances which led to considerable "gaps" in the tracking volume. As a result, the user cannot freely walk around for exploring the MRE but is instead bound to stay very close to targets. The user's movements are further as stronger motion blur needs to be avoided in order not to break the tracking.

Also the insufficient registration quality is mainly related to the insufficient video resolution and occurring motion blur causing error-prone tracking data. Other causes to name are imprecisions of the virtual reconstruction model as well as imperfectly installed and less trackable targets.

The prototypes results therefore underline the importance of a fully covered tracking volume as well as of highly accurate tracking data and room reconstruction.

Solutions to above imperfections are, for one, to use a camera providing higher resolution video. With the fast advance of mobile technology, the solution to this problem is a matter of time. The reliable retrieval of the virtual replica could be achieved by employing sophisticated 3D scanning methods.

Regarding the target quality, this factor is given by the conditions of the sites. It has already been stated in the analysis that sites might not provide a sufficient target density. It has now been established that potential targets additionally need to be of superior quality. Consequently, the probability that visual tracking can be successfully used for implementing the MRS in a site is therefore even lower. IrMoCap systems would hence provide a better solution, although also these might not applicable for any sites. Concluding, no universal tracking solution for the MRS could be found.

The prototype's performance, however, achieved stable real-time results and therefore meets the stated requirements. Although the MRE was equipped with abundant virtual objects and rich interactive behaviours, the computational latency remained at maximum 12.9 ms or 1.9 ms for rendering, respectively, which is only a fraction of the 40 ms (for 25 fps) maximum limit stated in the frame conditions. This result shows that the MRS could even successfully deliver scenarios of a far higher vertice count, finer textures and shaders and more complex interactive behaviours.

5 FINAL CONCLUSION

The goal of this thesis was to present a novel type of virtual heritage medium that immerses and engages users into the (re)-staging of events and situations related to the historical background of a site of cultural heritage. It was suggested to realise this by developing a mixed reality authoring framework based on Unity and an iPhone 5 that facilitates interactive and environment-related storytelling.

In order to present an effective and efficient solution approach, the technical requirements to this goal were analysed and the resources' technical constraints were assessed. Based on these analyses, a catalogue was presented that states the technical frame conditions necessary for achieving the targeted result. It was also assessed that the iPhone can only partly deliver the desired user experience for providing only a small display that does not allow the user to fully immerse into the MRE.

The catalogue of frame conditions formed the basis for the development process. A range of possibilities for designing the MRE and implementing the phone tracking was evaluated. It has been shown that Unity offers sufficient tools for creating MRE scenarios with time-linear or interactive storytelling elements. Regarding tracking solutions, infrared motion capture systems and natural feature based visual tracking were evaluated as most suitable. It has been stated, though, that the feasibility of either solution entirely depends on the conditions of the site:

Motion capture systems would deliver the required tracking quality but can only be used in sites that allow the installation of external set-ups. This solution therefore meets case B of the success classification scheme. Case A, an in-phone implemented tracking system, can only be implemented if a site provides a sufficient density of high-quality targets.

The visual tracking based prototype proved, though, that the necessary target density and quality for achieving permanent and stable tracking is unrealistic for any site. This is due to the limited iPhone camera resolution as well as its sensitivity to motion blur. With the given resources, visual tracking is hence not a feasible solution.

However, the prototype testing with a richly equipped example scenario also showed that the system could not only deliver permanent real time update rates but that even way more complex scenarios would be possible.

Concluding, it has been shown that the MRS can be implemented with the computing capacities of an iPhone 5 but can only in combination with a IrMoCap system. Therefore, case B applies: The system works but requires external set-ups. If visual tracking is to be employed, other hardware possibilties for providing sufficient camera resolutions should be taken into consideration.

6 APPENDIX A
STILLS FROM THE PROTOTYPE TEST RUN

Fig. A-1: View of the MRE 1

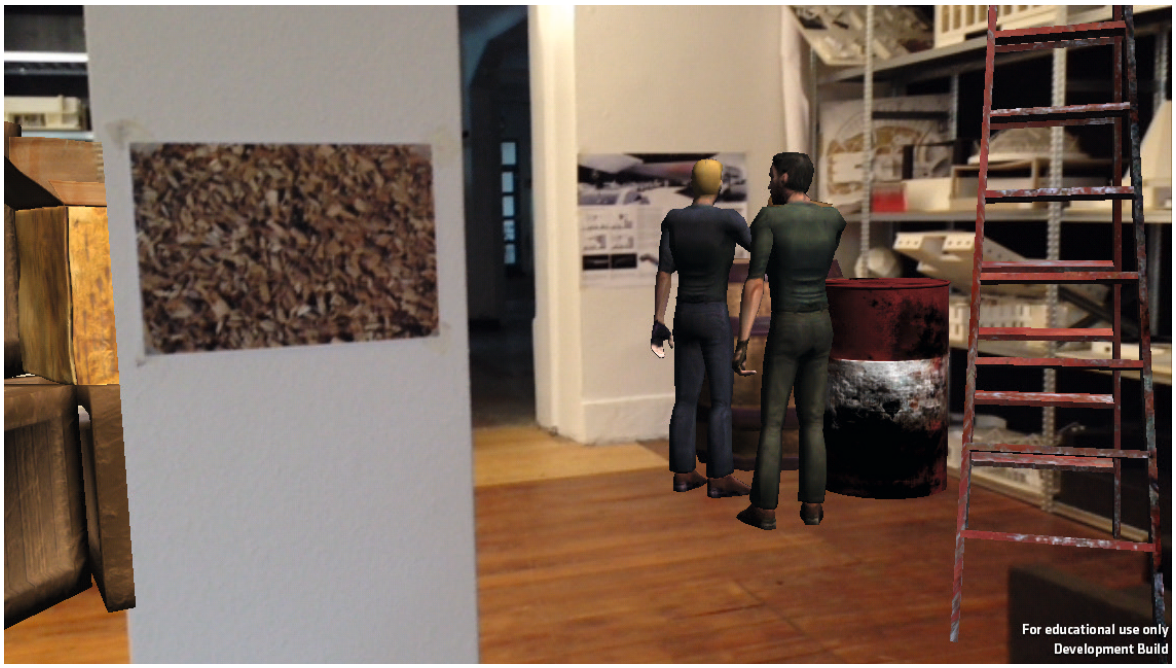

Fig. A-2: View of the MRE 2

Fig. A-3: View of the MRE 3

Fig. A-4: Characters walking

Fig. A-5: Character interaction 1

Fig. A-6: Character interaction 2

Fig. A-7: Characters working

7 APPENDIX B – ENCLOSURES

This thesis is enclosed a DVD-ROM containing

- a PDF version of this thesis
- the prototype project files (Unity Pro 4.3.2f1)
- a demo video (Quick time) of the prototype character behaviour in the Unity scene (file B0)
- videos of iPhone screen captures from the prototype test run (QuickTime, files B1 - B12). Please note: The videos occasionally show freeze frames which did not occur during the test run but are recording errors of the screen capture software.
- photographs of the test site (files B13 - 15)
- full-resolution screenshots from the prototype test run (B16 - B37)

8 BIBLIOGRAPHY

Abawi, D., Jose Luis Los Arcos, Michael Haller, et al.
2004 A Mixed Reality Museum Guide: The Challenges and Its Realization. http://www.researchgate.net/publication/228992359_A_mixed_reality_museum_guide_The_challenges_and_its_realization/file/d912f50981d6f8a559.pdf, accessed November 7, 2013.

Aitenbichler, Erwin, Fernando Lyardet, Aristotelis Hadjakos, and Max Mühlhäuser
2009 Fine-Grained Evaluation of Local Positioning Systems for Specific Target Applications. *In* Ubiquitous Intelligence and Computing. Daqing Zhang, Marius Portmann, Ah-Hwee Tan, and Jadwiga Indulska, eds. Pp. 236–250. Lecture Notes in Computer Science. Springer Berlin Heidelberg. http://dx.doi.org/10.1007/978-3-642-02830-4_19.

An, Bin
2012 Benchmarking the Accuracy of Inertial Sensors in Cell Phones. Master's thesis in electrical engineering, University of California. http://escholarship.org/uc/item/9z80c174.pdf, accessed January 7, 2014.

Apple
2013a Technical Specifications iMac. http://www.apple.com/imac/specs/, accessed January 25, 2014.
2013b Technical Specifications iphone 5. http://www.apple.com/iphone-5c/specs/, accessed January 26, 2014.

Apple Developer Library
2013 AV Foundation Programming Guide. https://developer.apple.com/library/mac/documentation/AudioVideo/Conceptual/AVFoundationPG/Articles/00_Introduction.html, accessed January 26, 2014.

Azuma, Ronald T.
1997 A Survey of Augmented Reality. Presence 6(4): 355–385.

Bahl, Paramvir, and Venkata N. Padmanabhan
2000 RADAR: An in-Building RF-Based User Location and Tracking System. *In* Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies Proceedings Pp. 775–784. New York: IEEE Computer Society. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=832252, accessed February 6, 2014.

Basset, Carlton
2013 A Mathematical Look at Focal Length and Crop Factor. PetaPixel. http://petapixel.com/2013/06/15/a-mathematical-look-at-focal-length-and-crop-factor/, accessed March 17, 2014.

Bevan, Nigel
2001 Human-Computer Interaction Standards. International Journal of Human-Computer Studies 55(4): 533–552.

Bregler, Christoph, Clothilde Castiglia, Jessica DeVincezo, et al.
2005 Squidball: An Experiment in Large-Scale Motion Capture and Game Design. *In* Intelligent Technologies for Interactive Entertainment Pp. 23–33. Berlin: Springer. http://link.springer.com/chapter/10.1007/11590323_3, accessed January 8, 2014.

Cambridge University Computer Laboratory
2005 The Bat Ultrasonic Location System. http://www.cl.cam.ac.uk/research/dtg/attarchive/bat/, accessed November 20, 2013.

Cawood, Stephen
2007 Augmented Reality: A Practical Guide. Pragmatic Programmers. Raleigh, N.C: Pragmatic Bookshelf.

Champion, Erik
2002 Cultural Engagement in Virtual Heritage Environments with Inbuilt Interactive Evaluation Mechanisms. *In* Proceedings of the Fifth Annual International Workshop Pp. 117–128. Porto: Universidade Fernando Pessoa. http://www.temple.edu/ispr/prev_conferences/proceedings/2002/Final%20papers/Presence2002-all%20papers.pdf.

Chandler, Jim, and John Fryer
2011 Accuracy of AutoDesk 123D Catch? Aboriginal Cave Re-Measurement Using Digital Photogrammetry. http://homepages.lboro.ac.uk/~cvjhc/otherfiles/accuracy%20of%20123dcatch.htm, accessed February 5, 2014.

CyArk
2013 CyArk-Projects. http://archive.cyark.org/project-world, accessed February 5, 2014.

DiVerdi, Stephen, and Tobias Hollerer
2007 Groundcam: A Tracking Modality for Mobile Mixed Reality. *In* Virtual Reality Conference Proceedings Pp. 75–82. New York: IEEE. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4161008, accessed November 8, 2013.

Egges, Arjan, George Papagiannakis, and Nadia Magnenat-Thalmann
2007 Presence and Interaction in Mixed Reality Environments. The Visual Computer 23(5): 317–333.

Frei, Erwin, Jonathan Kung, and Richard Bukowski
2005 High-Definition Surveying (HDS): A New Era in Reality Capture. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences 36: 204–208.

Furht, Borko
2011 Handbook of Augmented Reality. Berlin: Springer.

Garmin
N.d. What Is GPS? http://www8.garmin.com/aboutGPS/, accessed February 6, 2014.

Gauglitz, Steffen, Tobias Höllerer, and Matthew Turk
2011 Evaluation of Interest Point Detectors and Feature Descriptors for Visual Tracking. International Journal of Computer Vision 94(3): 335–360.

Hoff, William A., Khoi Nguyen, and Torsten Lyon
1996 Computer-Vision-Based Registration Techniques for Augmented Reality. *In* Proceedings of SPIE Pp. 538–548. Intelligent Robots and Computer Vision XV: Algorithms, Techniques,Active Vision, and Materials Handling. Washington, DC, USA: SPIE. http://proceedings.spiedigitallibrary.org/proceeding.aspx?articleid=1025038, accessed November 8, 2013.

Iuppa, Nicholas V.
2006 Story and Simulations for Serious Games: Tales from the Trenches. Amsterdam ; Boston: Elsevier.

Jesse James, Garrett
2011 The Elements of User Experience. California: New Riders Pub.

Kacyra, Ben, dir.
2011 Ancient Wonders Captured in 3D. TED.com. http://www.ted.com/talks/lang/de/ben_kacyra_ancient_wonders_captured_in_3d.html, accessed February 5, 2014.

Klein, Georg, and Tom Drummond
2003 Robust Visual Tracking for Non-Instrumental Augmented Reality. *In* Proceedings of the Second IEEE and ACM International Symposium on Mixed and Augmented Reality Pp. 113–122. New York: IEEE. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1240694, accessed November 8, 2013.

Kyle, Brock
2012 How Does the iPhone GPS System Work. http://www.everymac.com/systems/apple/iphone/iphone-faq/iphone-gps-definition-assisted-gps-how-gps-works-real-time-navigation.html, accessed February 6, 2014.

Law, Effie Lai-Chong, Arnold POS Vermeeren, Marc Hassenzahl, and Mark Blythe
2007 Towards a UX Manifesto. *In* Proceedings of the 21st British HCI Group Annual Conference on People and Computers Pp. 205–206. Swinton: British Computer Society. http://dl.acm.org/citation.cfm?id=1531468, accessed January 21, 2014.

Lima, João Paulo, Francisco Simões, Lucas Figueiredo, and Judith Kelner
2010 Model Based Markerless 3D Tracking Applied to Augmented Reality. SBC Journal on 3D Interactive Systems, 1(1): 2–15.

Lintermann, Bernd, Johannes Degenhard, Jan Gerigk, Martin Schmidt, and Manfred Hauffen
2011 Traffic. Augmented Reality Installation. ZKM Karlsruhe. http://www02.zkm.de/car-culture/index.php/de/werke/54-bernd-linterman, accessed November 7, 2013.

Liu, Hongbo, Yu Gan, Jie Yang, et al.
2012 Push the Limit of WiFi Based Localization for Smartphones. *In* Proceedings of the 18th Annual International Conference on Mobile Computing and Networking Pp. 305–316. New York: ACM. http://dl.acm.org/citation.cfm?id=2348581, accessed November 9, 2013.

Loomis, Jack M., James J. Blascovich, and Andrew C. Beall
1999 Immersive Virtual Environment Technology as a Basic Research Tool in Psychology. Behavior Research Methods, Instruments, & Computers 31(4): 557–564.

MacIntyre, Blair, Alex Hill, Maribeth Grandy, Brian Davidson, and Kimberly Spreen
N.d. UART. VPRN Wrapper. https://research.cc.gatech.edu/uart/content/about, accessed February 9, 2014.

Maes, Pattie
1995 Artificial Life Meets Entertainment: Lifelike Autonomous Agents. Communications of the ACM 38(11): 108–114.

Magnenat-Thalmann, Nadia, Marlène Arévalo, and George Papagiannakis
N.d. LIFEPLUS – Revival of Life in Ancient Pompeii. *In* Proceedings of the 8th International Conference on Virtual Systems and Multimedia Pp. 25–27. New York: IEEE. http://archiveweb.epfl.ch/vrlab.epfl.ch/Projects/projects_index.html.

Mann, Steve
2002 Mediated Reality with Implementations for Everyday Life. Presence Connect. http://wearcam.org/presence-connect/, accessed November 7, 2013.

Mann, Steve, and Woodrow Barfield
2003 Introduction to Mediated Reality. International Journal of Human-Computer Interaction 15(2): 205–208.

Martin, Eladio, Oriol Vinyals, Gerald Friedland, and Ruzena Bajcsy
2010 Precise Indoor Localization Using Smart Phones. *In* Proceedings of the International Conference on Multimedia Pp. 787–790. New York: ACM. http://dl.acm.org/citation.cfm?id=1874078, accessed November 5, 2013.

Matt, Halenka
2013 Voxel Metric | 3D Scanning Technology Overview: Kinect Reconstruction Algorithms Explained. http://voxelmetric.com/3d-scanning-technology-overview-kinect-reconstruction-algorithms-explained/, accessed February 5, 2014.

Maya User's Guide
2013 Bump Maps. http://download.autodesk.com/global/docs/maya2014/en_us/index.html?url=files/Surface_Relief__Displacement_maps.htm,topicNumber=d30e632545, accessed March 8, 2014.

Merienne, Frédéric
2010 Human Factors Consideration in the Interaction Process with Virtual Environment. International Journal on Interactive Design and Manufacturing (IJIDeM) 4(2): 83–86.

Metaio Developer Portal
2013 Tracking Configuration. http://dev.metaio.com/sdk/tracking-config/, accessed February 14, 2014.

Microsoft Kinect
2014 Kinect Fusion. http://msdn.microsoft.com/en-us/library/dn188670.aspx, accessed February 5, 2014.

Milgram, Paul, and Fumio Kishino
1994 A Taxonomy of Mixed Reality Visual Displays. IEICE TRANSACTIONS on Information and Systems 77(12): 1321–1329.

Mosaker, Lidunn
2001 Visualising Historical Knowledge Using Virtual Reality Technology. Digital Creativity 12(1): 15–25.

Naimark, Michael
1991 Elements of Real-Space Imaging: A Proposed Taxonomy. *In* SPIE/SPSE Electronic Imaging Proceedings Pp. 169–179. Washington, DC, USA: SPIE.

Nielsen, Jakob
2001 10 Heuristics for User Interface Design. http://www.nngroup.com/articles/ten-usability-heuristics/, accessed January 7, 2014.

Noguera, José M., and Juan C. Torres
2012 Interaction and Visualization of 3D Virtual Environments on Mobile Devices. Personal and Ubiquitous Computing 17(7): 1485–1486.

Pankratz, Frieder
2009 Tracking Für Augmented Reality Anwendungen Auf Mobiltelefonen. Master's thesis in informatics, TU Munich.

De la Peña, Nonny, Peggy Weil, Joan Llobera, et al.
2010 Immersive Journalism: Immersive Virtual Reality for the First-Person Experience of News. Presence: Teleoperators and Virtual Environments 19(4): 291–301.

Peng, Chunyi, Guobin Shen, Zheng Han, et al.
2007 A Beepbeep Ranging System on Mobile Phones. *In* Proceedings of the 5th International Conference on Embedded Networked Sensor Systems Pp. 397–398. New York: ACM. http://dl.acm.org/citation.cfm?id=1322313, accessed November 11, 2013.

Peng, Chunyi, Guobin Shen, and Yongguang Zhang
2012 BeepBeep: A High-Accuracy Acoustic-Based System for Ranging and Localization Using COTS Devices. ACM Transactions on Embedded Computing Systems 11(1): 1–29.

Peternier, Achille, Xavier Righetti, Mathieu Hopmann, et al.
2007 Chloe@ University: An Indoors, HMD-Based Mobile Mixed Reality Guide. http://www.peternier.com/research/publications/pdf/vrst2007.pdf, accessed November 8, 2013.

Phone Arena
2013 Apple iPhone 5 Full Specs. http://www.phonearena.com/phones/Apple-iPhone-5_id7378, accessed January 25, 2014.

Prasolova-Forland, Ekaterina, Mikhail Fominykh, Ramin Darisiro, and Anders I. Morch
2013 Training Cultural Awareness in Military Operations in a Virtual Afghan Village: A Methodology for Scenario Development. *In* System Sciences (HICSS), 2013 46th Hawaii International Conference on Pp. 903–912. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6479941, accessed November 10, 2013.

Preece, Jennifer, Yvonne Rogers, and Helen Sharp
2002 Interaction Design beyond Human Computer Interaction. Danvers: John Wiley & Sons.

Pressigout, Muriel, and Eric Marchand
2007 Real-Time Hybrid Tracking Using Edge and Texture Information. The International Journal of Robotics Research 26(7): 689–713.

Qiu, Jian, David Chu, Xiangying Meng, and Thomas Moscibroda
2011 On the Feasibility of Real-Time Phone-to-Phone 3d Localization. *In* Proceedings of the

9th ACM Conference on Embedded Networked Sensor Systems Pp. 190–203. http://dl.acm.org/citation.cfm?id=2070962, accessed November 9, 2013.

Qualcomm Vuforia Developer Portal
2013a Image Target Enhancement Tricks. https://developer.vuforia.com/resources/dev-guide/image-target-enhancement-tricks, accessed February 14, 2014.
2013b Natural Features and Rating. https://developer.vuforia.com/resources/dev-guide/natural-features-and-rating, accessed February 14, 2014.
2013c Projection Matrix Explained. https://developer.vuforia.com/resources/dev-guide/projection-matrix-explained, accessed March 18, 2014.

Rabbi, Ihsan, Sehat Ullah, and Siffat Ullah Khan
N.d. Augmented Reality Tracking Techniques: A Systematic Literature. http://www.researchgate.net/publication/235769903_Augmented_Reality_Tracking_Techniques_A_Systematic_Literature_Review_Protocol/file/9fcfd51363aef0691b.pdf, accessed November 5, 2013.

Sanchez-Vives, Maria V., and Mel Slater
2004 From Presence towards Consciousness. *In* 8th Annual Conference for the Scientific Study of Consciousness. Thorveton: Imprint Academic. http://www.researchgate.net/publication/7933300_From_presence_to_consciousness_through_virtual_reality/file/e0b495219d75949cfc.pdf, accessed January 15, 2014.

SAPOS
N.d. dGPS in BaWü. http://www.sapos-bw.de/dienste.php.

Seeber, Günter, and Martin Schmitz
N.d. Methodik Der GPS- Und DGPS-Messung. http://gio.uni-muenster.de/beitraege/ausg96_1/schmitz_meth_gps/iwu-pap3.htm, accessed November 9, 2013.

Skogstad, Stale A., Kristian Nymoen, and Mats Høvin
2011 Comparing Inertial and Optical Mocap Technologies for Synthesis Control. *In* Proceedings of the International Sound and Music Computing Conference Pp. 421–426. Padova: Padova University Press. http://smcnetwork.org/system/files/smc2011_submission_124.pdf, accessed February 6, 2014.

Slater, Mel
2009 Place Illusion and Plausibility Can Lead to Realistic Behaviour in Immersive Virtual Environments. Philosophical Transactions of the Royal Society B: Biological Sciences 364(1535): 3549–3557.

Sommer, Andreas
2013 Metaio SDK & Unity3D - Multiple Marker Types - One Scene. Metaio Helpdesk. http://helpdesk.metaio.com/questions/14862/metaio-sdk-unity3d-multiple-marker-types-one-scene, accessed February 14, 2014.

Soulard, Christopher E., and Rian C. Bogle
2011 Using Terrestrial Light Detection and Ranging (Lidar) Technology for Land-Surface Analysis in the Southwest. USGS. http://pubs.usgs.gov/fs/2011/3017/fs2011-3017.pdf, accessed February 5, 2014.

Stirton, Laura
2013 Protecting Art From UV Light. ARTcare. http://artcareinc.com/Protecting_Art_From_UV_Light.html, accessed January 26, 2014.

Stockyard Hill Wind Farm, ed.
2009 Parameters of Human Vision and Viewshed Definition. Environmental Resources Management Australia. http://www.stockyardhillwindfarm.com.au/pdf/PPAR_Annexes/ATS/Annexes/Annex_J/AnnexJ-LVA_PART_12.pdf.

Strand, Tor Egil Riegels
2008 Tracking for Outdoor Mobile Augmented Reality: Further Development of the Zion Augmented Reality Application. Master's thesis in computer science, Norwegian University of Science and Technology. http://ntnu.diva-portal.org/smash/record.jsf?pid=diva2:348709, accessed November 8, 2013.

Tan, BENG-KIANG, and HAFIZUR Rahaman
2009 Virtual Heritage: Reality and Criticism. *In* Proceedings of the 13th International CAAD Futures Conference Pp. 143–156. Montréal: Les Presses de l'Université de Montréal. http://cumincad.architexturez.net/system/files/pdf/cf2009_143.content.pdf, accessed November 10, 2013.

Taylor II, Russell M., Thomas C. Hudson, Adam Seeger, et al.
2001 VRPN: A Device-Independent, Network-Transparent VR Peripheral System. *In* Proceedings of the ACM Symposium on Virtual Reality Software and Technology Pp. 55–61. New York: ACM. http://dl.acm.org/citation.cfm?id=505019, accessed February 9, 2014.

The Hobbit: Production Diary, vol.13
2013 The Hobbit. Production Diary. New Zealand. http://www.youtube.com/watch?v=25dVQU3JkkE&feature=youtube_gdata_player, accessed February 6, 2014.

The ICOMOS Ename Charter for the Interpretation of Cultural Heritage Sites
2008 International Journal of Cultural Property 15: 377–383.

TomTom
2013 Portable GPS Car Navigation Systems. http://www.tomtom.com/howdoesitwork/page.php?ID=8&CID=2&Language=1, accessed February 6, 2014.

Tost, Laia Pujol, and Erik Malcolm Champion
2007 A Critical Examination of Presence Applied to Cultural Heritage. *In* The 10th Annual International Workshop on Presence Pp. 245–256. http://www.temple.edu/ispr/prev_conferences/proceedings/2007/Tost%20and%20Champion.pdf, accessed January 21, 2014.

Unity Documentation
2013a Light. https://docs.unity3d.com/Documentation/Components/class-Light.html, accessed March 6, 2014.
2013b Rigidbody. http://docs.unity3d.com/Documentation/Components/class-Rigidbody.html, accessed March 8, 2014.
2013c Draw Call Batching. https://docs.unity3d.com/Documentation/Manual/DrawCallBatching.html, accessed March 7, 2014.
2013d Light Probes. https://docs.unity3d.com/Documentation/Manual/LightProbes.html, accessed March 7, 2014.
2013e Occlusion culling. hhttps://docs.unity3d.com/Documentation/Manual/OcclusionCulling.html, accessed March 8, 2014.

Vacchetti, Luca, Vincent Lepetit, and Pascal Fua
2004a Stable Real-Time 3d Tracking Using Online and Offline Information. Pattern Analysis and Machine Intelligence, IEEE Transactions on 26(10): 1385–1391.
2004b Combining Edge and Texture Information for Real-Time Accurate 3d Camera Tracking. *In* Proceedings of the IEEE International Symposium on Mixed and Augmented Reality (ISMAR) Pp. 48–56. New York: IEEE Computer Society. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1383042, accessed November 21, 2013.

VICON
2002 Essentials of Motion Capture. VICON Motion Systems Ltd. http://www.udel.edu/PT/Research/MAL/essentials_of_motion_capture_v1_2.pdf, accessed July 2, 2014.
2008 Information – Get Going with the VICON MX-T Series. VICON Motion Systems Ltd.
2013 VICON T-Series. VICON Motion Systems Ltd. http://www.VICON.com/content/fileupload/System/T-SeriesLR.pdf, accessed July 2, 2014.

Wang, Jih-fang, Ronald T. Azuma, Gary Bishop, et al.
1990 Tracking a Head-Mounted Display in a Room-Sized Environment with Head-Mounted Cameras. *In* Orlando'90 Pp. 47–57. Washington, DC, USA: SPIE. http://proceedings.spiedigitallibrary.org/proceeding.aspx?articleid=942844, accessed November 8, 2013.

Wang, Xiangyu, and Marc Aurel Schnabel
2008 Mixed Reality In Architecture, Design, And Construction. Berlin: Springer.

Ward, Andy
2005 Ubisense Location Platform Accuracy. Cambridge: Ubisense Limited.

Wiebe, Robert
2011 Unity iOS Essentials Develop High Performance, Fun iOS Games Using Unity 3D. Birmingham, U.K.: Packt Pub.

Wither, Jason, Rebecca Allen, Vids Samanta, et al.
2010 The Westwood Experience: Connecting Story to Locations via Mixed Reality. *In* Mixed and Augmented Reality-Arts, Media, and Humanities (ISMAR-AMH) Pp. 39–46. New York: IEEE. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5643295, accessed November 6, 2013.

Wuest, H., F. Vial, and D. Stricker
2005 Adaptive Line Tracking with Multiple Hypotheses for Augmented Reality. *In* Proceedings of the 4th IEEE/ACM International Symposium on Mixed and Augmented Reality Pp. 62–69. Washington, DC, USA: IEEE Computer Society.

Yilmaz, Alper, Omar Javed, and Mubarak Shah
2006 Object Tracking: A Survey. ACM Computing Surveys 38(4): 13–58.

Young, Alexander D.
2010 Wireless Realtime Motion Tracking System Using Localised Orientation Estimation. University of Edinburgh. http://www.google.de/

ADDTIONAL IMAGE SOURCES

Figure 1 Compositing with material from

3D character model - knight templar. http://th01.deviantart.net/fs70/PRE/f/2013/203/8/c/3d_character_model___knight_templar_by_macx85-d5mbrr2.jpg, accessed 23.03.2014;

3D character model - 13th century heraldic knight. http://th01.deviantart.net/fs71/PRE/f/2013/202/3/4/3d_character_model___13th_century_heraldic_knight_by_macx85-d6e6bk7.jpg, accessed 23.03.2014;

Helmsley Castle - Helmsley. http://www.hello-yorkshire.co.uk/images/details/43/helmsley-castle-exhibition.jpg, accessed 23.03.2014;

Rennessaince Princess. http://www.dreamstime.com/royalty-free-stock-photos-renaissance-princess-image27242468, accessed 23.03.2014.

iPhone5 in hand. http://make-lemonade.co/themes/landahoy/img/slide1.png, accessed 23.03.2014.

Figure 7 Stairs In The Castle. *http://hdw.eweb4.com, accessed 22.03.2014*